O AUTOR:

DELIO SANTOS LIMA é seguramente um dos maiores técnicos no Brasil, reconhecido internacionalmente, sobre microcomputadores com lógica Sinclair.

Escreveu e publicou, em menos de três anos, nada menos que sete livros em mais de vinte e cinco edições, incluindo a primeira publicação específica para micros TK, NEZ, etc...

"45 PROGRAMAS PRONTOS PARA RODAR"

É o autor da única coletânea de programas para Sinclair, genuinamente nacional, abrangendo as normas do IAPAS, Imposto de Renda e outras:

"APLICAÇÕES SÉRIAS PARA TK 85 E CP 200"

Publicou a mais completa coleção de jogos em um único volume:

"30 JOGOS PARA TK E CP 200"

que, com programas desde 2K, até programas de 16K, como um Jogo de Damas e outros em código, consagrou-se como o ABC dos jogos

No livro

"CÓDIGO DE MÁQUINA PARA TK E CP 200"

Ofereceu uma agradável surpresa para aqueles que não exploravam as reais capacidades do computador Sinclair por falta de um material completo e compreensível. Trouxe um curso do Assembly Z80, com muitos macetes da 8KROM, mais de dez programas, um Dicionário do Assembly Z80, além de comparar as ROMs dos ZX, TK e NEZ, etc...

Criou um livro de consulta rápida do Basic Sinclair

"DICIONÁRIO DO BASIC SINCLAIR"

onde dividiu e classificou as diferentes instruções, comandos, funções e operadores Sinclair e os definiu em ordem alfabética. Acresceu-o com os capítulos Contar e Economizar Memória, além de comparar o Basic Sinclair com os demais.

Não satisfeito com os periféricos existentes, mais uma vez inovou, publicando:

"HARDWARE SINCLAIR"

que oferece circuitos como RESET, FONTE, MONITOR E INVERSOR DE VÍDEO, EXPANSÕES de RAM E ROM PARA 2, 4, 8 e 16K, PORTAS IN-OUT, TECLADO, ESQUEMA DO MICRO, LAY-OUT DOS CIRCUITOS IMPRESSOS E Outros.

O limite é a imaginação. As possibilidades do micro ainda não foram de todo exploradas. Eis-:

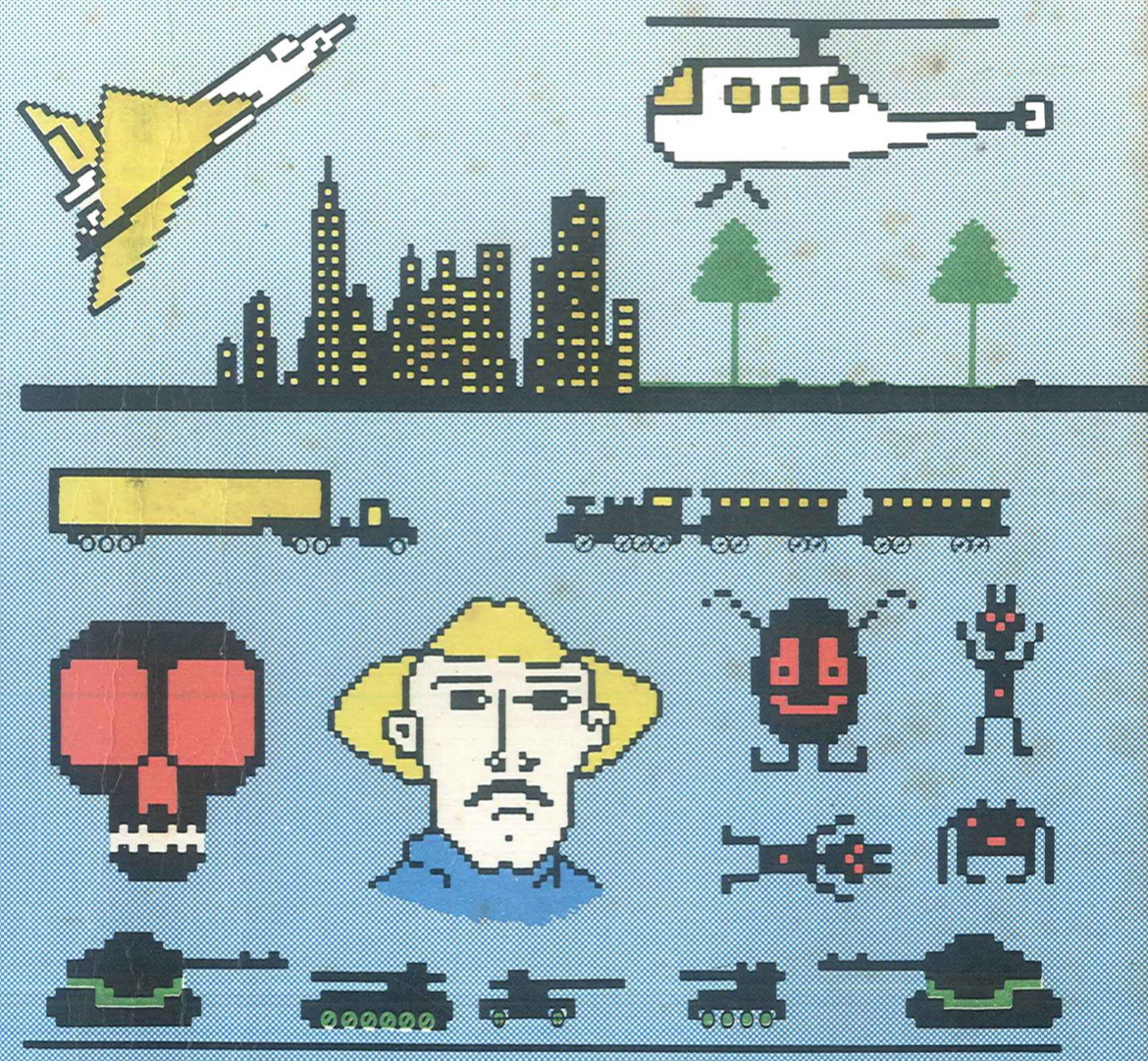
"200 DESENHOS PARA TK E CP 200"

com mais de 200 desenhos para jogos e ilustrações. Programas para edição de desenhos, em Basic e em Linguagem de máquina, rotinas para cópia instantântea do display, simulando movimentos, merging figuras, etc... Ensina as técnicas de vídeo, mapeamento, movimento, etc...

DELIO SANTOS LIMA

200 DESENHOS para TK e CP 200

200 DESENHOS para TK e CP200



PROGRAMA ENTER GRAPHICS EM BASIC E CÓDIGO
ROTINAS PARA MOVIMENTO, CÓPIA, GRAVAÇÃO INSTÂNTANEA DO VÍDEO
MAIS DE DUZENTOS DESENHOS CODIFICADOS PARA DIGITAÇÃO DIRETA

Délio Santos Lima

Do mesmo autor

45 PROGRAMAS PRONTOS PARA RODAR EM TK 82C e NE
APLICAÇÕES SÉRIAS PARA TK 85 E CP 200
30 JOGOS PARA TK E CP 200
CÓDIGO DE MÁQUINA PARA TK E CP 200
DICIONÁRIO DO BASIC SINCLAIR
HARDWARE SINCLAIR
SINCLAIR 8K ROM DISSASSEMBLY COMPLETO COMENTADO

200 DESENHOS TK 85 E CP 200
por Délio Santos Lima
com rotinas adicionais de Gino Douglas de Carvalho

1ª Edição Abril de 1985

Editado e distribuído por Micron Software Ltda.
São José dos Campos — SP.

Impresso por J.A.C. Editora Ltda.

TODOS OS DIREITOS RESERVADOS

Nos termos da Lei que resguarda os direitos autorais, é proibida a reprodução total ou parcial, ainda que em sistemas similares, de qualquer forma ou por qualquer meio gráfico, fotográfico, microfílmico, reprográfico, fonográfico e videográfico. Vedada a memorização e/ou a recuperação total ou parcial em qualquer sistema de processamento de dados e a inclusão de qualquer parte da obra em qualquer programa comercializável.

A violação dos direitos autorais é punível como crime (art. 184 e paragráfos do Código Penal, conforme Lei Nº 6.895 de 17.12.80) com pena de prisão e multa conjuntamente com busca e apreensão e indenizações diversas — arts. 122, 123, 124 e 126 da Lei nº 5.988, de 14.12.73, Lei dos Direitos Autorais.

PREFÁCIO

O objetivo deste volume não é ensinar a nobre arte do desenho. Pretendo é transmitir conceitos para a codificação de desenhos e figuras existentes em imagem de microcomputador.

Evidentemente far-se-ão necessários alguns conhecimentos sobre o que é possível codificar, a memória do vídeo, programas que facilitam a edição de imagens, que simulam movimento, etc...

Aqui você encontrará tudo isto e uma vasta coleção de desenhos, já codificados, de forma a facilitar ao máximo a sua utilização. Vá em frente. Pinte o sete.

Delio Santos Lima
Caixa Postal 100
12200 São José dos Campos
SP - Brasil

ÍNDICE

PARTE I — UMA INTRODUÇÃO

Introdução 09
A organização da memória 11
A memória do vídeo 13
PLOT & UNPLOT 15
Traçando linhas e círculos 17
Figuras geométricas 20
Figuras trigonométricas 23
Figuras simétricas em arte grega 27

PARTE II — O PROGRAMA ENTER GRAPHICS EM BASIC 31

Enter graphics em Basic 33
Usando as figuras só com Basic 43
Exemplo A — Movimento de lábios 44
Exemplo B — Mensagem com movimento de lábios 45
Exemplo C — Geografia. Aula, teste ou jogo com os mapas 47
Exemplo D — Geometria. aula, teste ou jogo com as figuras geométricas 49

PARTE II — O PROGRAMA ENTER GRAPHICS EM CÓDIGO 51

Enter graphics em código 53
Merging das figuras (usando-as em outros programas já existentes) 58
Listagem 1 — Programa para carregar o ENTERGRAPHICS 59
Listagem 2 — Os códigos do enter Graphics 61
Listagem 3 — Os códigos do Merging figuras 64
Usando as figuras com linguagem de máquina 66

Exemplo A — Animação 69
Exemplo B — Os invasores em tanques, naves, aviões e trens 69

PARTE IV — OS 200 DESENHOS 71

ATENÇÃO

Alterações a serem feitas a esta edição;

pág.	linha	encontra-se:	altere para:
05	26	64	65
34	19	420	435
43	11	766	770
45	25/26		1115 CLS
54	20	PEEK 1651 *PEEK	PEEK 16511+256*PEEK
54	21	1240	1242
58	26	32500	32558
64	01	LISTAGEM 3	Cont.da listagem 2.
65	10	4082	LISTAGEM 3 4082
66	18	3 rotinas	5 rotinas
66	23	160	220
18	14	horizontais	verticais
18	22	AT 20,C;	AT C,20;"■"
18	24	verticais	horizontais
19	6	AT L,30;	AT 20,L;"■"
19	25	12*SIN X	12*COS X
20	21	12*SIN X	12*COS X
21	2		60 NEXT N
21	19	10 FOR N=20 TO 45	10 FOR N=20 TO 30
21	20	70 PLOT N,45	30 PLOT 45,N 70 PLOT N,30
22	6	100 NEXT N	50 FOR N=20 TO 45 100 NEXT X
22	12	50 LET X=24	50 LET N=24

E. & O.E.

PARTE I

UMA INTRODUÇÃO

INTRODUÇÃO

Como já dizia aquele velho provérbio chinês: "Melhor do que dar um peixe a quem tem fome, é ensinar-lhe a pescar".

Por mais ilustrações e desenhos que este livro tivesse, certamente você ainda poderia querer algo diferente. O melhor mesmo é passar todas as informações de como desenhar no vídeo do micro, restringindo as figuras de exemplos a cerca de trezentas.

É bom ter sempre em mente as limitações da máquina para evitar decepções futuras. Não será possível desenhar imagens com mais resolução do que o vídeo do sistema. Aqui usaremos até 24 linhas de 32 caracteres sendoque cada posição do vídeo no modo gráfico pode ser subdividida em 4 simulando um total de 3.072 posições.

Pode-se ainda obter uma resolução de 768X8X8 pontos no vídeo, porque cada caractere é formado por oito pontos em 8 linhas de TV, resultando em 49.152 pontos. Isto requer a alteração do hard ware, será de processamento lento, além de consumir uma quantidade de memória de que o sistema não dispõe. E por isto não será alvo deste volume.

Tendo apenas 3.072 posições de vídeo, devemos considerar que:

1. Não poderemos representar com precisão imagens com grande número de formas, pelo menos, não de uma só vez.
2. Fotografias e outras imagens com meios-tons são mais difíceis de representar que desenhos a nanquim, como por exemplo; personagens de estórias em quadrinhos.
3. Quanto mais linhas de uma tela forem usadas, mais memória de "vídeo" para o vídeo será usada.

Não é aconselhável tentar criar movimentos em grandes imagens a partir da memorização de diferentes imagens, como em desenho animado e sim por alteração da imagem memorizada.

Não, pelo menos em Basic. Por isto oferecemos o programa ENTER GRAPHICS em uma versão na linguagem da máquina com a opção de animação. Funciona pela cópia instantânea das 12 imagens da área das variáveis para o vídeo, simulando movimento. Veja o exemplo da página 69.

A ORGANIZAÇÃO DA MEMÓRIA

O microprocessador Z80 dos micros Sinclair possue 16 linhas (binárias) de endereços podendo identificar 65536 (de 0 a 65535) posições de memória (de 8 bits cada). Bit é apenas abreviatura de Binary digiT, i. e., digito binário.

Segundo critérios do construtor, a memória dos computadores Sinclair foi distribuída como a seguir:

ENDEREÇO	UTILIZAÇÃO
00 000	sistema operacional (8KROM)
08 192	reserva para sistema operacional
16 384	variáveis do sistema operacional (RAM)
16 509	programa do usuário
(D. File)	vídeo
(Vars.)	variáveis do programa do usuário
(E. Line)	linha em edição
(StkBot)	stack do calculador
(StkEnd)	área livre
(SP)	stack da máquina
(ErrSP)	stack para endereços de sub-rotinas
(RamTop)	último endereço da memória em uso

Os endereços da tabela acima que não estão definidos, é porque apresentam-se em posições variáveis da memória devido às alterações do número de bytes de memória usada em cada uma destas áreas.

A ROM está nos endereços de 00 000 a 08 191.

A RAM está nos endereços de 16 384 em diante.

Como foi mencionado acima, a área da memória destinada ao usuário, situa-se nos endereços 16 509 em diante, existindo, após a mesma, uma série de outras áreas reservadas para diferentes propósitos. Estas áreas apresentam-se com comprimento ou consumo de memória variável. Suas posições na memória variam em função das variações de comprimento do programa em Basic, da RAM instalada, etc...

O endereço de início de cada uma destas áreas é arquivado pelo programa monitor (da ROM) na RAM do usuário nos endereços 16 384 a 16 508 e são as denominadas variáveis do sistema operacional. Não confundir com sistema de variáveis (do usuário).

Os endereços em que se encontram estas variáveis estão descritos no manual do seu equipamento. Citamos abaixo os mais usuais.

(D. File)	(16 396)
(Vars.)	(16 400)
(E. Line)	(16 404)
(RamTop)	(16 389)

Todas as variáveis do sistema mencionadas neste capítulo ocupam 2 bytes. Para se ler um valor armazenado em dois bytes usa-se a regra:
PRINT PEEK E+256*PEEK(E+1) onde E é o endereço dado.

Para entender por que o valor do segundo byte é multiplicado por 256 leia o capítulo BINÁRIO E HEXADECIMAL do livro CÓDIGO DE MÁQUINA PARA TK E CP 200 até a página 16 onde se lê: "A razão: é a diferença dos índices das potenciais do 2º byte".

A MEMÓRIA DO VÍDEO

Denominada pelos projetistas da máquina de D. File (Display File), a memória do vídeo não se encontra em endereços fixos e sim flutuando no topo da memória usada para o sistema de variáveis e programa. Usa a RAM do usuário.

O vídeo com 24 linhas de 32 caracteres correspondem a 24 X 33 bytes na memória do vídeo. Uma linha ocupa 33 bytes, sendo 32 caracteres e 1 para o término da linha (New Line ou Enter).

Cada caractere da tela ocupa um determinado byte na memória.

Os endereços da memória de vídeo variam de tempos em tempos, por questões de FIRMWARE. O endereço de início, ou do primeiro byte da memória do vídeo (D. File) é dado pela variável do sistema D. File, endereços 16 396 e 16 397.

O D. File começa com o primeiro byte em valor 118 (New Line) e também termina por 118.

O programa, a seguir, lê os 32 primeiros bytes da memória do vídeo e inverte os caracteres, acrescentando 128 aos seus códigos. Rode preferivelmente em SLOW:

```
10 LET DFILE = PEEK 16396+256*PEEK 16397
20 PRINT "QUALQUER COISA"
30 REM INVERTER UMA LINHA DO VÍDEO
40 FOR N=1 TO 32
50 LETX=PEEK (D FILE + N)
60 IF X<>118 THEN POKE D FILE+N, X+128
70 NEXT N
```

A posição de impressão, ou endereço da memória de vídeo onde a próxima impressão será executada, pode ser identificada pela variável do sistema denominada PRINT POSITION ou CC, CORRENT CARACTER, endereço 16398.

As duas linhas inferiores do vídeo são reservadas para edição e inputs, mas podem ser usadas, desde que se altere a variável do sistema do endereço 16418, cujo valor é habitualmente dois. Esta alteração, por POKE 16418, 0, por exemplo, não funciona como comando direto, somente como linha de programa. Experimente:

```
10 POKE 16418, 0
20 FOR N=0 TO 23
30 PRINT AT N, N; "OK"
40 NEXT N
```

Atenção: A variável do endereço 16418 deve voltar ao valor 2 ou mais, antes de qualquer instrução INPUT ou o programa será "destruído".

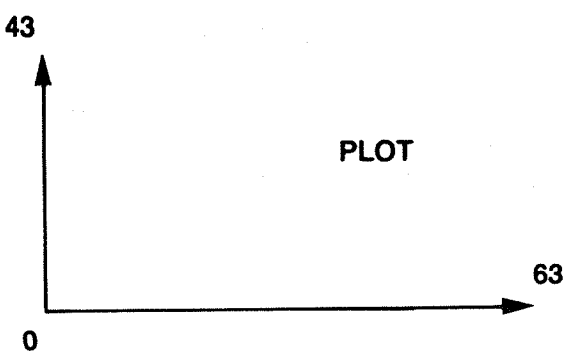
PLOT E UNPLOT

O Basic em 8KROM do Sinclair ZX81, TK 82,83,85, CP200, etc..., possui os comandos gráficos PLOT e UNPLOT. Considerando que o vídeo da máquina possui as posições representadas abaixo por quadrinhos (pixel), podemos assinalar ou apagar, qualquer um deles, desde que conhecidas as suas coordenadas (localização horizontal e vertical).

Cada quadrinho ou pixel possui sua localização horizontal numerada de 0 a 63 e vertical de 0 a 43.

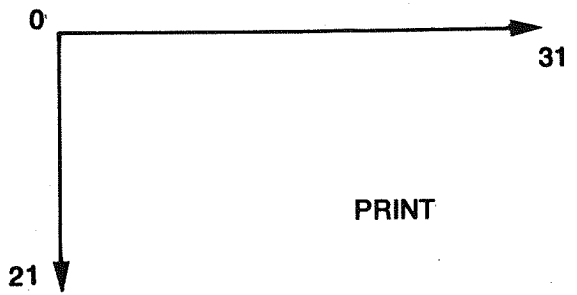
PLOT difere de PRINT AT por várias razões. Entre elas:

1. PLOT não requer e não permite a especificação do caractere a ser assinalado por utilizar-se de 1/4 do inverse space.
2. Existem duas vezes mais posições de PLOT na horizontal e na vertical resultando em quatro vezes mais resolução do que com PRINT.
3. A localização das linhas de PLOT é o inverso de PRINT. Veja abaixo.



As posições foram numeradas de baixo para cima ao invés de cima para baixo para facilitar a construção de gráficos e seus eixos.

A numeração das colunas se fez da esquerda para a direita assim como com PRINT AT.



Experimente como comando direto, sem número de linha.

```
PLOT 0,0
PLOT 0,43
PLOT 63,0
PLOT 63,43
```

RELEMBRANDO:

PLOT L,C linha coluna

Use PLOT L,C evita confusões.

PRINT e PRINT AT

Se você chegou até aqui sem conhecer bem PRINT e PRINT AT estamos mal. Leia no manual da sua máquina a descrição de funcionamento de PRINT e PRINT AT e releia este capítulo desde o início.
Não descrevemos aqui apenas para não "reinventar a roda ou o já inventado".

TRAÇANDO LINHAS E CIRCULOS

Nas horizontais e verticais não há qualquer mistério, nem grandes possibilidades quer com PLOT ou PRINT AT.
Quanto às linhas diagonais no vídeo Sinclair "alguns experts" criaram ou copiaram equações que não tem muitas soluções ou opções, senão vejamos:

LINHAS DIAGONAIS

As diagonais podem ser traçadas a partir de "horizontais deslocadas uma ou mais posições na vertical para cada posição de impressão deslocada na horizontal ou vice versa. O programa a seguir ilustra esta proposição:

```
10 SLOW
20 LET L=0
30 LET C=32
40 PLOT C,L
50 LET L=L+1
70 GOTO 40
```

Este programa como está gera uma linha vertical. Acrescente:

```
60 LET C=C+1 ou 60 LET C=C-1
```

que produzirá uma linha diagonal de + 45° ou -45°. Experimente:

```
60 LET C=C+1.5
60 LET C=C+2
60 LET C=C+3
etc...
```

Para abrir a área remanescente do vídeo volte para

```
60 LET C=C+1
```

e altere também

```
50 LET L=L+2 ou LET L=L+3
```

etc..., ou ainda use

```
60 LET C=C-1 ou 60 LET C=C-2 etc...
```

Para trabalhar a partir de uma horizontal basta inverter a linha 40 para:

```
40 PLOT L,C
```

Para ajustá-la ao centro da tela use

```
30 LET C=21
```

Para traçar diagonais com PRINT AT ou qualquer outro caracter-re o princípio é o mesmo. Basta alterar

```
40 PRINT AT C,L; ou 40 PRINT AT L,C;
```

e centrar a linha por

```
30 LET C=1 ou 30 LET LET C=16
```

LINHAS HORIZONTAIS

Escolhe-se qual a linha a ser traçada, de 0 a 43 para PLOT ou de 0 a 21 para PRINT AT. Escolhemos a 20.

```
10 FOR C=0 TO 63
20 PLOT 20,C
30 NEXT C
```

Ou então

```
10 FOR C=0 TO 20
20 PRINT AT 20,C;
30 NEXT C
```

LINHAS VERTICAIS

Escolhe-se qual a coluna a ser usada, de 0 a 63 para PLOT ou de 0 a 31 para PRINT AT.

Escolhemos a 30.


```
10 FOR L=0 TO 63
20 PLOT L,30
30 NEXT L
```

Ou então com PRINT AT

```
10 FOR L=0 TO 31
20 PRINT AT L,30;
30 NEXT L
```

CÍRCULOS

Os círculos podem ser facilmente desenhados com o uso das funções trigonométricas SEN e COS. Devido à pequena precisão do vídeo, os círculos ficam melhores, se apenas tracejados como com esta rotina:

```
10 FOR F=1 TO 60
20 LET X=F/5*PI
30 LET A=21+6*SIN X
40 LET B=22+6*COS X
50 PLOT A,B
60 NEXT F
```

O número de pontos por círculo é de 12, conforme a listagem dada. Para alterá-la para 20, insira:

```
20 LET X=F/10*PI
```

Para alterar o diametro do círculo para o dobro por exemplo, altere:

```
30 LET A=21+12*SIN X
40 LET B=21+12*SIN X
```

FIGURAS GEOMÉTRICAS

Figuras e formas geométricas podem ser desenhadas por programação com muito mais facilidades do que se entradas ponto a ponto da tela. A seguir alguns exemplos:

1. CÍRCULOS

Devido à pequena precisão do vídeo, os círculos ficam melhores se apenas tracejados como com esta rotina que utiliza as funções SIN e COS:

```
10 FOR F=1 TO 60
20 LET X= F/5*PI
30 LET A=21+6*SIN X
40 LET B=22+6*COS X
50 PLT A,B
60 NEXT F
```

O número de pontos por círculo é de 12, conforme a listagem acima. Para alterá-lo para 20, insirá:

```
20 LET=F/10*PI
```

Para alterar o diâmetro do círculo para o dobro por exemplo, altere:

```
30 LET A=21+12 * SIN X
40 LET B=21+12 * SIN X
```

2. QUADRADO

```
10 FOR N=20 TO 30
20 PLOT 19,N
30 PLOT N,20
40 PLOT 30,N
```



```
50 PLOT N,30
```

Para alterar o tamanho, alterar os valores 20 e 30.

Observar que a impressora TS 2040 deforma os caracteres ligeiramente, quanto ao formato quadrado apresentado no vídeo. Ou seja: um quadrado no vídeo pode resultar em um retângulo na impressora TS 2040, porque se cada um dos caracteres é alongado, a imagem toda o será também.

3. RETÂNGULO

Em pé:

```
10 FOR N= 20 TO 40
20 PLOT 20,N
30 PLOT 30,N
40 NEXT N
50 FOR N=20 TO 30
60 PLOT N,20
70 PLOT N,40
80 NEXT N
```

Para "deitar" o retângulo do programa anterior, altere nele:

```
10 FOR N=20 TO 45
70 PLOT N,45
```

4. TRIÂNGULO EQUILÁTERO

```
10 FOR N=21 TO 37
20 PLOT N,20
30 NEXT N
40 LET K=1
50 FOR N=20 TO 29
60 PLOT N,N
70 PLOT N+9,30-K
80 LET K=K+1
90 NEXT N
```

5. TRIÂNGULO RETÂNGULO

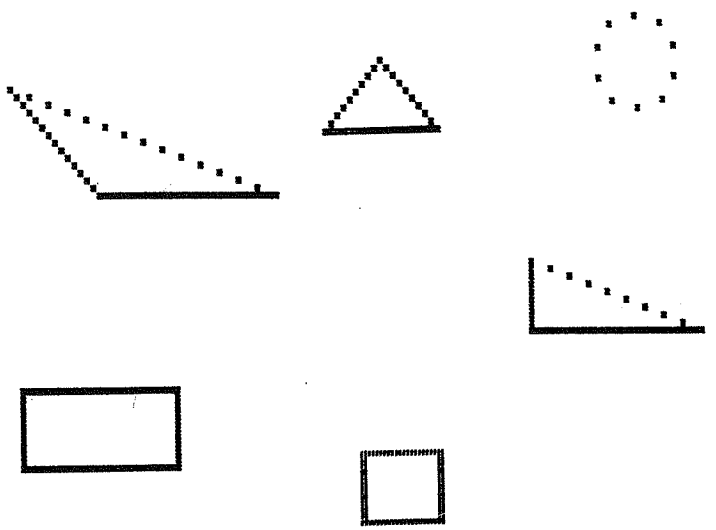
```
10 LET X=16
20 FOR N=25 TO 51
30 PLOT N,X
40 NEXT N
```

```
50 LET N=24
60 FOR X=16 TO 25
70 PLOT 25,X
80 PLOT 3*N-20,X
90 LET N=N-1
100 NEXT N
```

6. TRIÂNGULO OBTUSO

```
10 LET X=16
20 FOR N=24 TO 52
30 PLOT N,X
40 NEXT N
50 LET X=24
60 FOR X=16 TO 30
70 PLOT N,X
80 PLOT 3*N-20,X
90 LET N=N-1
100 NEXT X
```

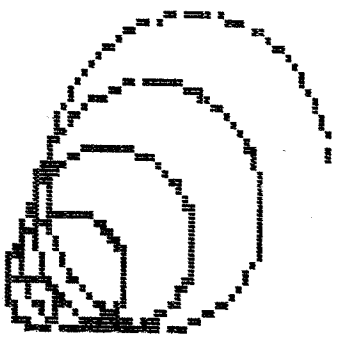
A seguir, as figuras produzidas por estes programas.



FIGURAS TRIGONOMÉTRICAS

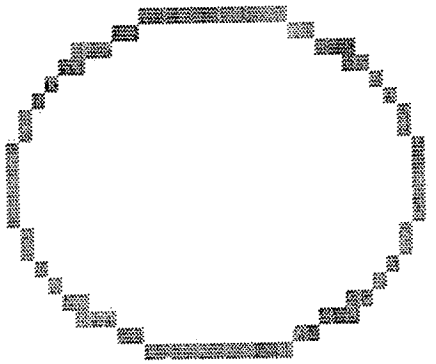
As rotinas básicas para traçarem os gráficos das funções trigonométricas estão nos manuais do equipamentos e em outros livros como Dicionário do Basic Sinclair e por isto aproveitamos a oportunidade para apresentar "outras formas" com trigonometria:

1. ESPIRAL CÔNICA



```
10 REM MICRON
20 FOR X=1 TO 500
30 LET Y=PI*X/50
40 LET Z=(500-X)/500
50 PLOT (20.5*COS (Y)+30)*Z,(2
0*SIN (Y)+20)*Z
60 NEXT X
```

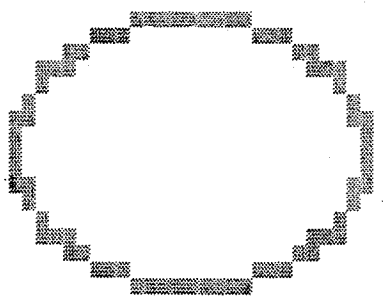
2. FORMA OVAL



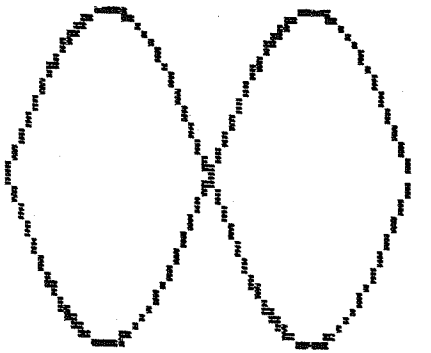
```
10 REM MICRON
20 FOR X=1 TO 120
30 LET Y=PI*X/50
40 PRINT AT 10*COS (Y)+11,15*S
IN (Y)+15;"▒"
50 NEXT X
```

Para diminuir o tamanho, altere:

```
40 PRINT AT 8*COS (Y)+11,13*SI
N (Y)+15;"▒"
50 NEXT X
```

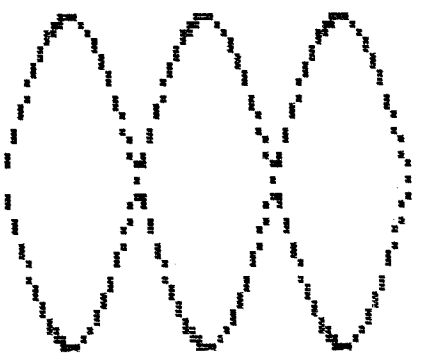


3. SIN+SIN INV



```
10 REM MICRON
20 FOR Z=1 TO 2
30 FOR X=1 TO 120
40 IF Z=1 THEN LET Y=X*PI/60
50 IF Z=2 THEN LET Y=-X*PI/60
60 PLOT X/2,SIN (Y)*20+20
70 NEXT X
80 NEXT Z
```

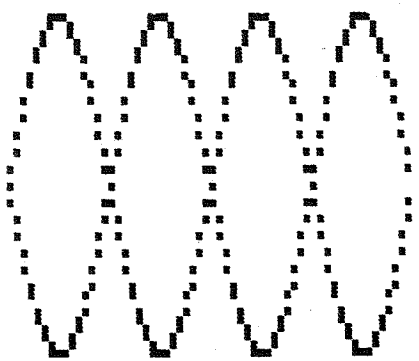
4. 1,5SIN+SIN INV



Altere

```
40 IF Z=1 THEN LET Y=X*PI/40
50 IF Z=2 THEN LET Y=-X*PI/40
```

5. 2 SIN+2SIN INV



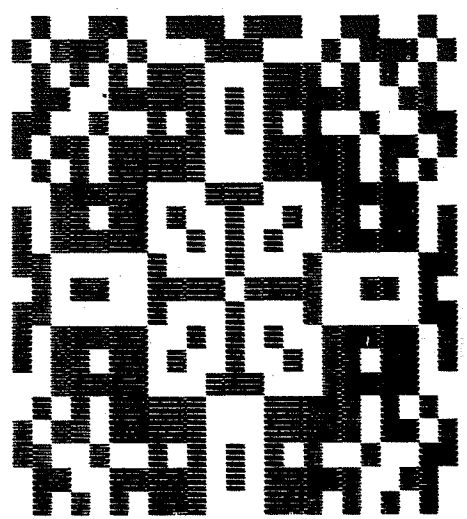
Altere

```
40 IF Z=1 THEN LET Y=X*PI/30
50 IF Z=2 THEN LET Y=-X*PI/30
```


FIGURAS SIMÉTRICAS

Uma infinidade de figuras semelhantes às apresentadas a seguir podem ser geradas por um único programa com pouco mais de dez linhas.

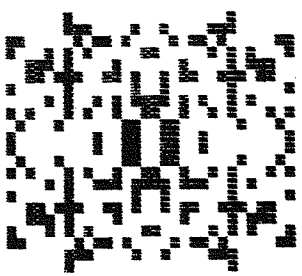
O bom efeito visual destas imagens é dado pela sua simetria e aleatoriedade. Utilizando-se do gerador de números randômicos o programa sorteia uma posição qualquer dentro de 1/4 da tela. Esta posição é também marcada em cada uma das outras partes de 1/4 da tela, equidistante das bordas dos seus respectivos lados opostos e o resultado é este:

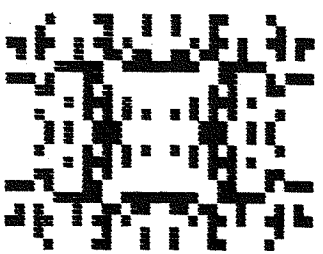


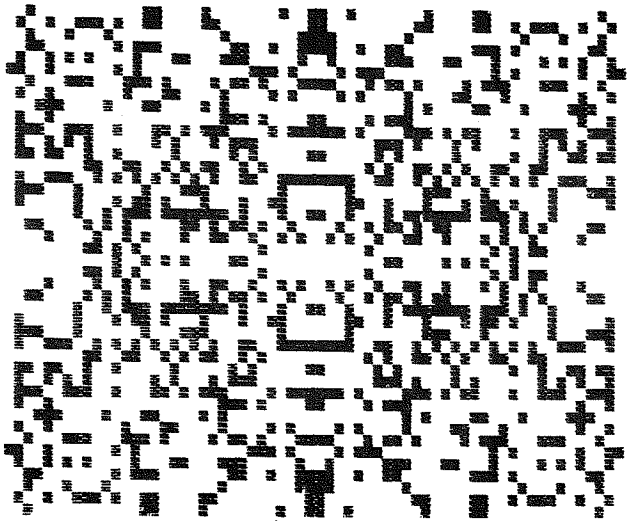
O programa utilizado:

```
  5 SLOW
 10 LET A=12
 15 LET A$="■"
 20 POKE 16418,0
 25 LET X=INT (RND*A)
 35 LET Y=INT (RND*A)
 40 PRINT AT A+X,A+Y;A$
 45 PRINT AT A+Y,A+X;A$
 50 PRINT AT A-X,A+Y;A$
 60 PRINT AT A+Y,A-X;A$
 70 PRINT AT A-X,A-Y;A$
 80 PRINT AT A-Y,A-X;A$
 90 PRINT AT A+X,A-Y;A$
100 PRINT AT A-Y,A+X;A$
110 LET X=X+RND*2-1
120 LET Y=Y+RND*2-1
130 IF RND<.3 OR ABS Y>A OR X>A
THEN GOTO 200
150 IF RND<.025 THEN GOTO 300
200 RUN
300 PAUSE 40000
310 CLS
320 RUN
```

O programa anterior trabalha com PRINT '■'. A seguir propomos outro programa similar utilizando PLOT que gera figuras tais como:







O **Programa**:

```
 10 REM MICRON
 20 SLOW
 30 LET A=INT (63*RND)
 40 LET B=INT (42*RND)
 50 IF RND>.5 THEN GOTO 150
 60 PLOT A,B
 70 PLOT A,42-B
 80 PLOT 63-A,B
 90 PLOT 63-A,42-B
100 RUN
110 UNPLOT A,B
120 UNPLOT A,42-B
130 UNPLOT 63-A,B
140 UNPLOT 63-A,42-B
150 RUN
```

PARTE II

ENTER GRAPHICS EM BASIC

ENTER GRAPHICS EM BASIC

Este programa visa facilitar a entrada de figuras com caracteres gráficos codificados ou entrá-las, desenhando com PLOT e UNPLOT ou PRINT com qualquer caractere. Armazena até 12 telas completas com 22 linhas. Possibilita alterações futuras, listagem das figuras, dos códigos usados, gravação, além de conter um sub-programa para carregar as rotinas em código como Merging, Scrool e outras que rodam o vídeo e podem fazer o Eduardo Busqueta dançar, a caveira falar ou os tanques e trens se moverem, escapando aos seus tiros.

DIGITANDO OS CARACTERES GRÁFICOS

Quando aparece uma figura identificar ou mesmo soletrar os caracteres usados mostra-se algo confuso, pois no exemplo dado poderiamos ter diferentes símbolos, resultando na mesma figura, como é o caso de e .

A identificação e leitura dos caracteres gráficos não é tão simples quanto com os alfanuméricos, mas para nossa felicidade Sir Sinclair colocou apenas um símbolo gráfico em cada tecla alfanumérica. Por isto basta simplesmente representarmos os caracteres gráficos pelas letras e números de suas respectivas teclas. Por exemplo: pode ser representado por 1 2 3 4 5 6 7.

Confira no teclado do micro.

O programa proposto, a seguir, transforma as letras e números digitados em seus equivalentes símbolos gráficos, comparando

as strings A$=" [illegible]"
e B$=" 12745TEADSPQW6R8Y3HGF"

Será a mesma coisa que digitar os caracteres gráficos sem precisar usar SHIFT-GRAPHICS, uma vez que as teclas serão as mesmas.

Utilizando-se dos caracteres alfanuméricos para a codificação dos gráficos, ficam as seguintes restrições:

A. Não se poderá usar nas figuras as letras e números que sejam "códigos" dos gráficos. Não é o caso de 0,Ø,X,V, etc... Também não serão usáveis o N, o P e o ponto.

B. Para usar os caracteres alfanuméricos como 0,Ø,X ou os inversos, eles devem estar incluidos ao final das strings A$ e B$. No programa listado as strings estão sem 0ØØØ. Para que, ao digitar 0 saia o inverso e Ø ídem, altere o final de A$ para [illegible]. Como estão, as strings A$ e B$ possuem 22 caracteres. Caso você altere o comprimento das mesmas será preciso alterar "o comprimento" do looping da linha 865 FOR T=1T026 que faz a conversão códigos para gráficos. Igualmente com a linha nº 420

C. Com SLOW não se pode digitar SPACE (espaço) diretamente, porque esta tecla atua como BREAK. Obtém-se espaço, digitando a tecla ao lado (ponto).

A tecla BREAK/SPACE atua como Break, quando o micro está em RUN e atua como SPACE no modo de edição ou durante INPUTS. Na versão SLOW o programa "roda" continuamente em um looping de INKEY$, enquanto que na versão FAST é INPUT variável.

O programa conforme está listado requer SLOW, devido o uso de INKEY$.

```
   1 REM XXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
   10 LET K=0
   25 CLS
   27 SLOW
   30 PRINT ,,,,"EDITOR DE VIDEO"
   35 PRINT ,,,,"PARA","DIGITE"
   40 PRINT ,,"APAGAR/INICIAR","1
 "
   42 PRINT "ALTERAR/CONT.","2"
   45 PRINT "LISTAR FIGURA","3"
```



```
  50 PRINT "LISTAR CODIGOS","4"
  60 PRINT "GRAVAR","5"
  70 PRINT "DESENHAR","6"
  80 INPUT X
  85 IF K=0 AND X<>1 THEN GOTO 8
0
  87 IF X=7 OR X=8 OR X>9 THEN G
OTO 80
  90 CLS
  95 GOTO X*100
 100 REM INPUT STRING
 102 PRINT "VAI APAGAR O ARQUIVO
..? DIGITE 1"
 103 INPUT N$
 104 IF N$<>"1" THEN GOTO 25
 115 CLS
 120 PRINT "ESTE ARQUIVO ACABA D
E SER DIMEN-SIONADO PARA 12 IMAG
ENS DE 22 LI"
 125 PRINT "NHAS COM 32 CARACTER
ES CADA UMA."
 128 PRINT ,,"PARA ALTERAR DIGIT
E BREAK E,","RESCREVA AS LINHAS
135 A 150."
 130 CLEAR
 135 DIM F$(12,22,32)
 140 LET K=1
 145 LET I=12
 148 LET C=32
 150 LET L=22
 152 DIM X$(32)
 154 LET K=1
 156 LET A$=" [illegible]
[illegible]"
 157 LET B$=" 12745TEADSPQW6R8Y3
HGF"
 158 LET C$="[illegible]
[illegible]"
 162 PRINT ,,"DIGITE UMA TECLA"
 163 LET N$=INKEY$
 164 IF N$="" THEN GOTO 163
 165 CLS
 166 FOR Z=1 TO I
 170 LET N=0
 172 LET N=N+1
 175 IF N=23 THEN GOTO 25
 176 GOSUB 800
 177 POKE 16418,2
 178 IF X$(1)="M" THEN GOTO 25
 179 IF X$(1)="C" THEN GOTO 185
 180 IF X$(1 TO 4)="COPY" THEN G
OTO 895
 184 GOSUB 855
```

```
185 PRINT AT 0+N-1,0;F$(Z,N)
195 IF N<23 THEN GOTO 172
196 CLS
197 NEXT Z
198 GOTO 25
200 REM ALTERA
202 PRINT "QUAL IMAGEM ALTERA O
U CONTINUA?"
204 INPUT Z
210 PRINT "QUAL LINHA ALTERA OU
CONTINUA?"
215 PRINT "DE 1 A ";L;"?"
217 INPUT N
220 CLS
224 PRINT AT 0,0;
225 FOR X=1 TO N
230 PRINT F$(Z,X)
235 NEXT X
240 GOTO 176
245 GOSUB 855
250 PRINT F$(Z,N)
260 PAUSE 3E4
270 GOTO 25
300 REM LISTA FIGURA
302 PRINT "QUAL A IMAGEM? DE 1
A ";I;"?"
304 INPUT Z
305 FAST
306 CLS
310 FOR N=0 TO 21
320 PRINT AT N,0;F$(Z,N+1)
330 NEXT N
335 SLOW
370 LET K$=INKEY$
377 IF K$="C" THEN COPY
380 IF K$="P" THEN GOTO 390
385 IF K$="M" THEN GOTO 25
387 GOTO 370
390 LET Z=Z+1
392 GOTO 306
405 FAST
410 PRINT "QUAL A IMAGEM?"
420 INPUT Z
422 CLS
425 FOR N=1 TO L
430 FOR B=1 TO C
432 LET X$=F$(Z,N)
435 FOR T=1 TO 22
440 IF X$(B)=A$(T) THEN PRINT B
$(T);
445 NEXT T
450 NEXT B
455 PAUSE 200
```



```
459 NEXT N
460 SLOW
470 GOTO 370
500 REM SAVE
510 PRINT ;;;;"ROTINA P/ SAVE"
520 PRINT ;;"DIGITE O NOME"
530 INPUT N$
540 PRINT "DIGITE    NEW LINE,"
550 PRINT "A SEGUIR LIGUE O CAS
SETTE."
560 INPUT L$
570 SAVE N$
580 GOTO 25
605 LET Z$="▇"
610 LET Z=1
615 LET X=0
620 LET Y=0
625 LET L$=INKEY$
630 GOSUB 690
640 IF L$="F" THEN GOTO 780
644 IF L$="C" THEN GOTO 700
650 LET Y=Y+(L$="7")-(L$="6")
655 LET X=X+(L$="8")-(L$="5")
660 GOSUB 680
662 IF X>63 THEN LET X=63
666 IF Y>43 THEN LET Y=43
668 PLOT X,Y
670 IF Z=0 THEN UNPLOT X,Y
672 GOTO 625
680 IF X<0 THEN LET X=0
682 IF Y<0 THEN LET Y=0
686 RETURN
690 IF L$="E" THEN LET Z=1
692 IF L$="A" THEN LET Z=0
694 RETURN
705 LET L$=""
710 UNPLOT X,Y
720 LET L$=INKEY$
725 GOSUB 690
735 IF L$="P" THEN GOTO 625
737 IF L$="F" THEN GOTO 780
740 LET X=X+(L$="6")-(L$="7")
745 LET Y=Y+(L$="8")-(L$="5")
750 GOSUB 680
755 IF X>31 THEN LET X=31
760 IF Y>21 THEN LET Y=21
762 IF L$="C" OR L$="A" OR L$="
E" OR L$="P" OR L$="F" THEN GOTO
775
770 IF CODE L$>37 THEN LET Z$=C
HR$ ((CODE L$)+128)
771 IF L$="." THEN LET Z$=" "
775 PRINT AT X,Y;Z$
776 IF Z=0 THEN PRINT AT X,Y;"
"
```

```
 778 GOTO 720
 780 FAST
 781 LET L=1
 782 LET DF=PEEK 16396+256*PEEK
16397
 783 LET N=1
 784 LET C=1
 785 FOR F=1 TO 726
 787 IF C=33 THEN LET L=L+1
 788 IF C=33 THEN LET C=0
 789 IF L=23 THEN GOTO 25
 790 IF (PEEK (DF+F)<>118) THEN
LET F$(N,L,C)=CHR$ (PEEK (DF+F))
 791 LET C=C+1
 792 NEXT F
 794 GOTO 25
 800 LET Z$=""
 801 SLOW
 802 POKE 16418,0
 804 PRINT AT 22,0;"IMAGEM N° ";Z
,N;"° LINHA"
 806 PRINT AT 23,0;C$
 808 PRINT AT 23,0;
 810 LET K$=INKEY$
 812 IF K$="." THEN LET K$=" "
 815 IF K$="" THEN GOTO 810
 817 IF K$="N" THEN GOTO 841
 818 IF CODE K$=118 THEN RETURN
 820 LET Z$=Z$+K$
 825 LET X$=Z$
 835 PRINT K$;
 837 LET YU=RND+RND
 840 GOTO 810
 841 LET LEN=LEN Z$
 842 PRINT AT 23,LEN-1;"■";AT 23
,LEN-1;
 843 IF LEN Z$>1 THEN LET Z$=Z$(
1 TO (LEN-1))
 844 LET X$=Z$
 845 GOTO 837
 855 REM CONVERTE
 857 FAST
 860 FOR B=1 TO C
 865 FOR T=1 TO 22
 870 IF X$(B)=B$(T) THEN LET F$(
Z,N,B)=A$(T)
 880 NEXT T
 885 NEXT B
 890 RETURN
 895 COPY
 896 LET X$=""
 897 PRINT AT 21,0;X$
 898 COPY
 899 GOTO 176
```


O MENU

1- APAGAR INICIAR

Após digitar e gravar o programa, comece com RUN e opção 1. Esta rotina irá apagar todo o arquivo já existente, se houver. Por isto, antes é pedida uma confirmação.

O número de imagens a serem armazenadas está préfixado em 12, mas pode ser alterado. Quanto maior o número de imagens, maior será o consumo de memória e o tempo de gravação. Por exemplo: Sem rodar o programa ou após CLEAR, ele ocupa ±3 minutos de gravação. Para 6 imagens ±5 minutos e para 12 imagens ±8 minutos. Veja o Enter Graphics em código, opção 5.

Para alterar o número de imagens para 6 por ex., altere as linhas

```
135 DIM F$(06,22,32)
145 LET I=6
```

A entrada de cada imagem é feita linha por linha.
No vídeo aparecerá:

```
IMAGEM N° 1                    1° LINHA
0123456789012345678901234567890 1
```

Impresso nas linhas 22 de 23. Linhas normalmente usadas para o cursor (edição e Inputs). Isto é obtido pelo comando da linha 802. Se você quiser usar este "truque" em outros programas, observe que, antes de qualquer Input, deve-se aplicar o comando da linha nº 177.

A linha que apresenta os números inversos é a linha usada para edição. Cada caractere entrado irá aparecer nela, na sua posição. Isto é, coluna 0, 1, 2 e assim por diante. Isto facilita a localização e a identificação do caractere em digitação. O término da linha é dado por New Line (Enter) e não é preciso completar o final com espaços.

Não use **Space** ou ocorrerá BREAK.

Para produzir espaço digite ponto (tecla ao lado do SPACE).

Para se repetir uma tecla, basta mantê-la presa por algum tempo.

A entrada é feita com o uso de INKEY$. A entrada pode ser mais rápida, alterando-se o retardo de tempo da linha 837 LET YU=RND**RND para 837 LET YU=RND*RND*RND que não será

tão lento quanto RND**RND.

Caso você digite algum caractere errado, não adianta tentar apagá-lo por DELETE (Shift-Ø) ou usar as facilidades de edição como cursor left, cursor right, porque não se está em um comando de INPUT ou em EDIÇÃO e sim em RUN com o uso de INKEY$.

Para apagar digite N e o caractere à esquerda será apagado. Para corrigir entre a seguir o novo caractere. É só.

Se o final da linha for espaços, não é preciso digitá-los. Digite New Line (Enter) e aguarde. Deverá demorar uns 30 segundos na execução da sub-rotina de conversão alfanuméricos X gráficos, até que seja impressa a 1ª linha de símbolos gráficos da figura.

Para sair desta rotina sem completar as 22 linhas, digite uma linha igual a M (Menu).

Para pular a linha em edição sem apagá-la ou ter que reescrevela, digite uma linha igual a C (CONT).

Caso acidentalmente você tenha acionado BREAK, até que se acostume usar o ponto como espaço, **Não** use **RUN** que apagará as variáveis (imagens). **USE GOTO 25 que é o menu** e continue na opção 2.

RESUMO DOS COMANDOS

TECLA	USO OU SIGNIFICADO
PONTO	Espaço
N<	Apaga o caractere a esquerda
Space/Break	Break
M + New Line	Goto Menu
C + New Line	Continua na linha seguinte
New Line	Final da linha
1 2 3 4 5 6 7 8 Q W E R T Y A S D F G H	O respectivo símbolo gráfico contido na tecla

2- ALTERAR/CONT

Quando acionada, esta rotina pede o número da imagem e o número da linha a ser alterada ou, de onde se quer continuar.

A seguir a imagem indicada é listada até a linha indicada, continuando na rotina de entrada de dados, opção 1 do menu. A alteração é feita, reescrevendo-se a linha em questão. Ela será impressa exatamente sobre a linha anterior (caso de erro) e a figura no vídeo ficará também corrigida.

Nesta rotina tem-se as opções de edição da opção 1, pois continua naquela.

3- LISTAR FIGURA

As figuras são armazenadas na matriz F$(12,22,32).

São até 12 figuras de 22 linhas com 32 caracteres cada.

PRINT F$(1,2) imprime a primeira linha da primeira imagem.
PRINT F$(1,2) imprime a segunda linha da primeira imagem.
PRINT F$ (2,1) imprime a primeira linha da segunda imagem.
PRINT F$ (1,1,1) imprime o primeiro caractere da primeira linha da primeira imagem.

Para se executar a impressão de uma imagem completa, com 22 linhas será preciso o looping a seguir:

```
7000 FOR N=1 TO 12
7010 PRINT F$(1,N)
7020 NEXT N
```

Este exemplo irá imprimir as 22 linhas da imagem 1.
A opção 3 do menu faz justamente isto, perguntando-lhe qual a imagem a listar.

Ao final da listagem, digite:
M para Menu
C para Copy (impressora)
P para próxima imagem

Caso você queira imprimir a imagem por um comando direto do tipo PRINT I$, sem usar o looping, use antes:

```
7000 DIM I$(710)
7010 FOR N=1TO 22
7020 LET I$=I$+F$(1,N)
7030 NEXT N
```

A seguir, PRINT I$ imprimirá toda a imagem.

4- LISTAR CÓDIGOS

Os códigos ou caracteres alfanuméricos que representam os gráficos não são armazenados. Logo após a edição de uma linha os códigos são transformados em seus equivalentes símbolos gráficos e armazenados na matriz F$(I,L,C).

Caso lhe interesse, os códigos de uma imagem podem ser recuperados e listados por esta opção, que pede o número da imagem a ser descodificada, ou melhor, codificada.

A listagem dos códigos é feita pela conversão dos símbolos gráficos armazenados em F$(I,L,C), em processo inverso ao da opção 1.

Esta rotina é lenta, ainda que rodada em FAST. Após a conversão de cada linha, esta é mostrada.

Ao final, digite:

M para Menu
C para Copy (impressora)
P para próxima listagem. Continua a listar a próxima figura e não os códigos.

5- GRAVAR

Esta rotina permite gravar o programa e suas variáveis (imagens). Continua no menu após a gravação ou após a carga.

6- DESENHAR

Esta rotina permite desenhar diretamente no vídeo do micro usando o pixel de PLOT e UNPLOT ou qualquer caractere com PRINT AT.

A imagem ao final é armazenada na variável F$ para posterior recuperação e impressão.

Os usos e aplicações desta rotina são totalmente distintos e diferentes da opção 2.

As facilidades de edição de "Desenhar no Vídeo" são:
1. A impressão começa na posição 0,0 com **PLOT**
2. Para comutar para PRINT Caractere digite C. Indica com inverse space "■".

3. Para voltar para PLOT digite P.
4. Para deslocar a posição de impressão sem escrever, digite A de Apagar. Vale igualmente com PRINT e PLOT.
5. Para voltar a escrever digite E. Vale igualmente durante PRINT e PLOT.
6. O deslocamento da posição de impressão é dado pelas teclas → ↓ ← ↑
7. Durante PRINT, para mudar de caractere, digite o desejado, excluindo-se os de comando (C,P,A,E,F e.). O caractere digitado aparecerá no vídeo em sua forma inversa. Para retirar esta inversão retire o + 128 existente à linha de número 766. Para voltar a inverse space NÃO digite SPACE,porque ocorrerá BREAK. Use o "." que voltará ao inverse space "■".
Caso, por esquecimento, você acione BREAK, NÃO digite CONT que inclui CLS e apagará o vídeo. Use GOTO à linha indicada no canto inferior do vídeo, ao lado do código de reportagem de BREAK 0/
8. Ao término do desenho, digite F para armazenar a imagem na variável F$.
9. Para recuperar e imprimir a imagem use a opção 3 do MENU.

USANDO AS FIGURAS

Se você já carregou ou pretende carregar alguma figura com o uso do programa Editor, eis algumas das possibilidades de aplicação da(s) mesma(s):

1- Simplesmente como estão, em Basic, listando-as no programa Editor ou criando programas a partir delas, como por exemplo: aula de geografia ou jogos com os mapas, figuras que "falam" enquanto uma mensagem é impressa, estórias em quadrinho, tapeçaria, figuras geométricas, trigonometria, etc... Sempre com movimentos lentos ou pequena área de movimento.

2- Com rotinas em código, inseridas no programa Editor, para produzirem movimentos rápidos como: cópia instântanea do vídeo, scrool para a direita ou para a esquerda e outras que podem fazer os personagens dançarem, os tanques e trens se moverem. Com o auxílio das rotinas no código da máquina, obtem-se movimentos rápidos de toda a imagem.
As figuras inseridas com o programa Editor podem ser transportadas para outros programas já previamente escritos, tanto por uma rotina em Basic como no código da máquina.

USANDO AS FIGURAS SÓ COM BASIC

A seguir propomos alguns exemplos práticos de utilização das imagens em Basic, como movimento (de lábios) com ou sem mensagem impressa, aula de Geografia, de Geometria, etc...

Para jogos e movimentos você, inevitavelmente, deverá usar o ENTER GRAPHICS na versão em código, bem como as suas rotinas adicionais, também em código.

A utilização destes programas não requer maiores conhecimentos além do Basic.

EXEMPLO A — MOVIMENTO DE LÁBIOS

Propaganda de Escolas de Dança, Computação, Línguas, Aviação, etc... usando respectivamente as figuras do Eduardo Busqueta, do Humanóide, do Charles Chaplin, do Santos Dumont, etc...

Como o Basic é um tanto lento, não vamos tentar impor grandes movimentos às imagens. Se nos limitarmos apenas a mover os lábios, teremos velocidade até demais,sendo necessário um retardo de tempo, como você verá.

Conforme a ilustração da página 77, veremos que os lábios do Eduardo Busqueta (parte inferior) estão AT 16,16; com os caracteres gráficos das teclas 7 e 1 que são ▬▬.

Para fazê-lo falar, basta alternarmos a impressão dos caracteres ▬▬ com os ▬▬ , com algum retardo de tempo.

ATENÇÃO

As coordenadas AT 16,16; não correspondem ao elemento (16,-16) da matriz F$, porque os elementos da matriz se iniciam em 1 (um) e as coordenadas de PRINT AT se iniciam em em 0 (zero).

A posição da tela PRINT AT 16,16 corresponde ao elemento (17,17) da matriz, ou seja: F$ (I,17,17) e F$ (I,17,18) são elementos da matriz F$ que correspondem aos lábios inferiores do nosso personagem. I é o número da imagem.

Isto é apenas para você lembrar. Por ora vamos usar diretamente no vídeo PRINT AT 16,16; ▬▬ e PRINT AT 16,16; ▬▬ após a impressão da matriz F$.

Acrescente ao programa ENTER GRAPHICS, versão BASIC, a rotina abaixo:

```
   1 REM NUNCA USE RUN - USE GOT
O 1
   2 GOTO 1000
1000 PRINT "QUAL A IMAGEM ?"
1010 INPUT I
1020 FOR N=1 TO 22
1030 PRINT F$(I,N)
1040 NEXT N
1060 FOR N=1 TO 10E4
1070 PRINT AT 16,16;"▬ ";
1080 LET YU=RND**RND
1090 PRINT AT 16,16;"▬ "
1135 NEXT N
```

EXEMPLO B — MENSAGEM COM O MOVIMENTO DE LÁBIOS

Esta continuação da rotina anterior vai fazer com que o Eduardo Busqueta fique falando (movendo os lábios) com a mensagem "DIGITE UMA TECLA" impressa na tela. Digitando-se uma tecla, não vale BREAK, aparecerá a primeira linha da mensagem e, a cada tecla pressionada, a mensagem será trocada até a terceira linha, repetindo-se.

A mesma coisa você poderá fazer com os outros personagens fornecidos.

A seguir, a listagem a ser acrescida ao programa "ENTER GRAPHICS", versão BASIC, com a rotina do exemplo A:

```
1115 CLS
1045 LET YX=0
1050 PRINT AT 21,0;"DIGITE UMA T
ECLA                    "
1100 LET N$=INKEY$
1110 IF N$="" THEN GOTO 1140
1112 LET YX=YX+1
1114 IF YX=2 THEN GOTO 1140
1116 IF YX=3 THEN GOTO 1160
1120 PRINT AT 21,0;"ISTO EH UMA
DEMONSTRACAO...         "
1125 GOSUB 1200
1135 GOTO 1100
1140 PRINT AT 21,0;"....SERIA DE
 UM COMPUTADOR ?        "
1150 GOSUB 1200
1155 GOTO 1100
1160 PRINT AT 21,0;"....OU DE BR
EAKDANCE ?              "
1165 GOSUB 1200
1170 GOTO 1045
1200 FOR T=1 TO 2
1210 NEXT T
1220 RETURN
```


EXEMPLO C — GEOGRAFIA

Utilizando-se mapas fornecidos ou outros de sua autoria, podemos facilmente transformá-los em aula, teste, jogo ou simples ilustrações.

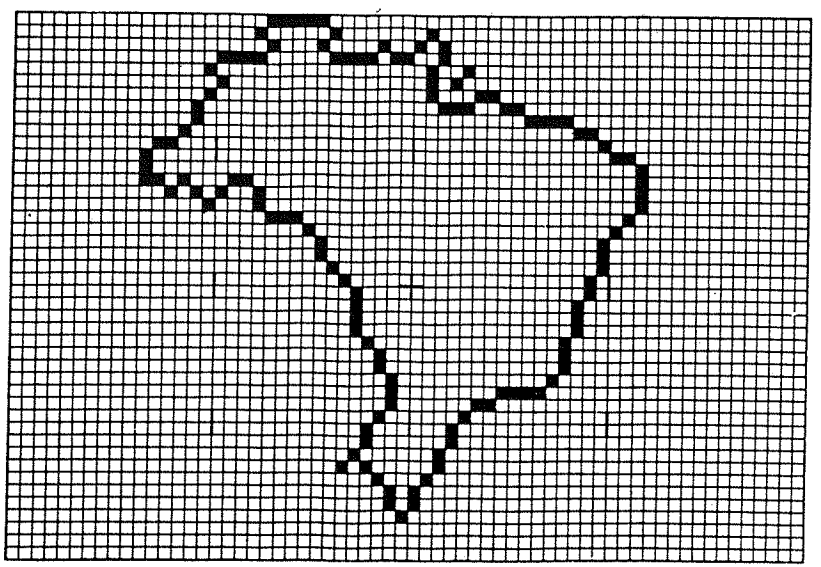
Por exemplo: Carregue com o ENTER GRAPHICS o mapa do Brasil e suas regiões na ordem abaixo:

IMAGEM Nº	DESCRIÇÃO
UM	BRASIL
DOIS	REGIÃO NORTE
TRÊS	REGIÃO NORDESTE
QUATRO	REGIÃO CENTRO-OESTE
CINCO	REGIÃO SUDESTE
SEIS	REGIÃO SUL

Acrescente à gravação do ENTER GRAPHICS com os mapas acima a rotina a seguir. É a simulação de um teste aleatório das regiões. Você pode, por exemplo, fazer a mesma coisa com os estados e as capitais ou com os continentes e os países.

```
   1 REM NUNCA USE RUN - USE GOT
O 1
   2 GOTO 1000
1000 PRINT "O BRASIL E SUAS REGI
OES"
1010 PRINT "VOU PERGUNTAR OS EST
ADOS DO BRASIL"
1020 PRINT "DIGITE UMA TECLA"
1030 LET N$=INKEY$
1040 IF N$="" THEN GOTO 1030
1050 CLS
1060 PRINT ".....UMA DEIXA,"
1070 PRINT ,,"ESTE EH O BRASIL R
EGIONAL ...."
1080 GOSUB 1600
1085 CLS
1090 FOR  N=1 TO 22
1100 PRINT F$(I,N)
1110 NEXT N
1130 GOSUB 1600
1150 DIM B$(6)
1155 LET B$(1)="BRASIL"
1160 LET B$(2)="NORTE"
1165 LET B$(3)="NORDESTE"
1170 LET B$(4)="CENTRO-OESTE"
```

```
1175 LET B$(5)="SUDESTE"
1180 LET B$(6)="SUL"
1200 CLS
1205 LET S=INT RND*5+2
1210 FOR N=1 TO 22
1220 PRINT F$(S,N)
1230 NEXT N
1240 PRINT AT 21,0;"QUAL E A REG
IAO ?"
1250 INPUT R$
1260 IF R$=B$(S) THEN GOTO 1300
1270 PRINT AT 21,0;"VOCE ERROU..
.."
1280 GOSUB 1600
1290 GOTO 1240
1300 PRINT AT 21,0;"PARABENS. E
A REGIAO ";R$
1310 GOSUB 1600
1320 GOSUB 1600
1330 GOTO 1200
1600 FOR T=1 TO 100
1610 NEXT T
1620 RETURN
```



EXEMPLO D — GEOMETRIA

As figuras geométricas básicas podem ser facilmente utilizadas em aulas ou testes.

Digite as rotinas do capítulo II — Figuras Geométricas numeradas, como a seguir, SEM o programa ENTER GRAPHICS.

ROTINA DO	NUMERAR DE		
QUADRADO	101	A	190
RETÂNGULO	201		290
CÍRCULO	301		390
TRIÂNGULOS EQUILÁTERO	401		490
TRIÂNGULO OBTUSO	501		590
TRIÂNGULO RETÂNGULO	601		690

A seguir acrescente:

```
   5 LET Z=INT (RND*6+1)
  10 GOSUB Z*100
  15 PRINT AT 18,0;"QUAL E A FIG
URA ?"
  20 INPUT X$
  25 IF X$<>N$ THEN GOTO 70
  30 PRINT AT 18,0;"PARABENS. E
O ";N$
  40 INPUT X$
  50 CLS
  60 GOTO 1
  70 PRINT AT 18,0;"VOCE ERROU.
TENTE OUTRA VEZ..."
  75 GOTO 20
 100 LET N$="QUADRADO"
 190 RETURN
 200 LET N$="RETANGULO"
 290 RETURN
 300 LET N$="CIRCULO"
 390 RETURN
 400 LET N$="TRIANGULO EQUILATER
O"
 490 RETURN
 500 LET N$="TRIANGULO OBTUSO"
 590 RETURN
 600 LET N$="TRIANGULO RETANGULO
"
 690 RETURN
```

PARTE III

ENTER GRAPHICS EM CÓDIGO

ENTER GRAPHICS EM CÓDIGO

Este programa, versão em código, foi especialmente concebido para este livro pelo Engenheiro Gino Douglas de Carvalho, autor do livro "40 Rotinas em Linguagem de Máquina".

Apresenta todas as facilidades de edição e armazenamento da versão em Basic associada à velocidade da Linguagem de Máquina, além das seguintes possibilidades inexistentes em Basic:

- grava as figuras independentemente do programa, assim como também as carrega;
- permite ainda a cópia instantânea das 12 figuras armazenadas nas variáveis para a tela, simulando movimento.

Tudo ainda pode ser usado com outras rotinas em código, como rodar o vídeo para a esquerda ou direita, inversor de vídeo, etc... fornecidas na listagem 4 ou com outras de sua autoria.

RESUMO DAS ROTINAS FORNECIDAS

LISTAGEM 1 — Programas em Basic para carregar os códigos (do ENTER GRAPHICS).
LISTAGEM 2 — ENTER GRAPHICS, em código.
LISTAGEM 3 — Programa em código para usar as figuras em outros programas, previamente escritos.
LISTAGEM 4 — Rotinas adicionais em código:
Simular movimento
Rodar o vídeo para a esquerda, para a direita
Scrool para cima em 1 coluna e
Inversor de Vídeo.

Este é um programa totalmente escrito em linguagem de máquina, possibilitando assim maior facilidade de Edição e uso das figuras deste livro. Para melhor aproveitarmos das vantagens resultantes do emprego desta linguagem, este programa tem funções diferentes da sua versão em Basic.

Antes de falarmos sobre o programa propriamente dito, vamos explicar como colocá-lo no computador. Sendo um programa relativamente longo, com 1240 Bytes, todo o cuidado será necessário na digitação, pois somente um Byte errado poderá por tudo a perder. Para facilitar, Micron Software comercializa esse programa em cassete através do pedido incluso no final deste volume. Se você observar a listagem 2, verificará que ela é formada por três partes principais: 1º) o endereço em hexadecimal (4082,408A,etc.); 2º) bytes dos códigos em si, também em hexadecimal; 3º) um número decimal corresponde ao check-sum dos 8 bytes anteriores.

A existência do check-sum é que nos facilitará o encontro de qualquer erro de digitação. Para iniciarmos devemos criar uma linha REM com pelo menos 1240 caracteres. Para verificarmos se esta linha está correta, basta digitar PRINT PEEK 1651*PEEK 16512 e se a resposta for1240 ou mais, estará correta.

Em seguida digitamos o programa da listagem 1, que é um monitor para a entrada do programa da listgem 2. Para iniciarmos a entrada de dados, digitamos RUN e, em seguida, o endereço desejado em Hexadecimal. Por exemplo, no início digitaremos 4082. Este endereço deverá ser digitado com os 4 digitos ou não será aceito. Em seguida este endereço aparecerá no canto inferior da tela e poderemos digitar os 8 bytes da listagem, sem espaços entre eles, por exemplo:

7676AF3222403255

Seguido de New-Line. Se não forem digitados os 16 caracteres eles não serão aceitos. Após alguns segundos aparecerá um número decimal após os bytes, devendo este número ser comparado ao check-sum. Se eles forem iguais, os 8 bytes digitados estarão corretos, podendo prosseguir no endereço seguinte, que já estará no canto inferior da tela.

Quando se está esperando a entrada dos 8 bytes, os seguintes comandos serão aceitos, chamados pela digitação do primeiro caracter e new-line:

S-STOP: Interrompe a execução do programa

R-Recomeça: volta a pedir o endereço, recomeçando a partir deste endereço.

E-ERRO: Caso o check-sum não esteja correto, digite este comando e o endereço anterior voltará a ser mostrado, devendo ser redigitados os 8 bytes.

Como você pode ver, não é necessário digitar todo o programa de uma só vez, pois podemos colocar os bytes a partir de qualquer endereço, sendo inclusive uma boa prática fazer varias gravações parciais conforme se vai digitando o programa. O único cuidado a tomar será sempre digitar os códigos nos endereços indicados (listagem 2) para que o check-sum tenha válidade.

Após totalmente montado, devemos deletar as linhas 10A a 1090, deixando sómente a linha REM e efetuar uma gravação antes do teste. Para podermos utilizar o programa, devemos reservar a área acima do RAMTOP para as doze figuras, através dos comandos: POKE 16388,0 e POKE 16389,94 seguindo-se por NEW e pela carga do programa do cassete.

Para rodarmos o programa utilizaremos o comando RAND USR 16516. Caso esteja tudo correto, aparecerá o MENU nas duas linhas inferiores da tela, conforme mostrado abaixo:

FIG VER-1 ALT-2 APG-3 LST-4 SV-5
0 LD-6 SVT-7 LDT-8 COP-9 AN-0

Sempre que estivermos no MENU bastará digitarmos NEW-LINE para voltar ao Basic. O comando RAND USR 16516 além de começar a execução também limpa a área acima do RAMTOP, onde ficarão as figuras. É indispensável que a primeira chamada seja feita através desta forma, para garantirmos que não existirão caracteres indesejáveis na área das figuras. Porém se tivermos feito várias figuras e, por um motivo qualquer, tivermos que retornar ao Basic, se utilizarmos RAND USR 16516, perderemos as figuras existentes. Devemos digitar o seguinte:

```
10 POKE 16418,0
20 RAND USR 16536
```

Com isso, entramos no programa sem apagar as figuras já existentes, porém não deve ser utilizado, assim que carregamos

o programa.

Vamos explicar agora as várias funções deste programa. Nós trabalhamos com o conceito de figura corrente, ou seja, a maioria das funções só terá efeito sobre a figura que estiver sendo mostrada no momento, que denominaremos de figura corrente. No canto esquerdo do MENU temos a seguinte identificação que mostra qual é a figura corrente:

FIG
0

No caso a figura corrente é a de número 0, podendo ser utilizadas 12 figuras, numeradas de 0 a 9, mais as figuras "A" e a figura "B". Para trocarmos de figura corrente utilizamos **a função 1 (VER-1).** Dititando-se 1 e um número válido de figura, esta figura passará a ser a figura corrente.

A função 2 (ALT-2) permite alterarmos a figura corrente, podendo inclusive modificá-la totalmente. É o comando que utilizaremos para a colocação de novas figuras. Quando digitamos 2 a partir do MENU, aparecerá a figura corrente na tela. Na parte inferior aparecerá: Linha:00 Coluna: 00, e no canto superior esquerdo aparecerá um G piscante, que é o cursor.

A indicação linha: 00 e coluna: 00 é a posição do cursor, podendo a coluna variar de 0 a 31, e a linha de 0 a 21. O "G" do cursor indica que estamos no modo Graphics. Se, por exemplo, digitarmos a tecla 1 aparecerá o caractere gráfico na posição 0,0 e o cursor estava na posição 0,1, conforme indicado abaixo. Note que não utilizamos a tecla Shift, e que existe auto repetição, assim como na versão em Basic.

O cursor pode ser mudado para "I", siginificando que podemos entrar com caracteres inversos, através de Shift 2, a voltarmos para "N", caracteres normais, através de Shift 1. Para voltarmos ao cursor "G", basta digitar Shift 9. Podemos movimentar o cursor através das teclas Shift 5, Shift 6, Shift 7 e Shift 8, sendo que o movimento do cursor, não é destrutivo, podendo passar sobre uma figura sem alterá-la. Para encerrarmos a Edição de uma figura devemos digitar New-Line e voltaremos ao MENU.

A função 3 (APG-3) permite que apaguemos a figura corrente, bastando digitarmos 3 e respondermos "S" à pergunta "quer realmente apagar (S/N)" que aparecerá na parte inferior do vídeo, retornando em seguida ao MENU. Se digitarmos "N" retornaremos ao MENU sem apagar a figura corrente.

A função 4 (LIST-4) permite que a figura corrente seja listada conforme seus códigos. Por exemplo o caractere gráfico será listado como a letra "Q". Os caracteres inversos não serão alterados e os caracteres normais não serão mostrados. Para retornar ao MENU digite qualquer tecla.

A função 5 (SV-5) permite gravar a figura corrente no cassete. Após digitarmos 5, aparecerá uma mensagem solicitando que seja preparado o gravador e pressione qualquer tecla para iniciar a gravação. Esta tem os 5 segundos usuais de silêncio no início, seguidos de aproximadamente de 15 segundos da gravação da figura propriamente dita. Após o SAVE retorna ao MENU.

A função 6 (LD-6) permite que façamos o lado de uma figura gravada no cassete com a função 5 (SV-5), colocando-a na figura corrente, apagando o seu conteúdo anterior. Aparecerá a mensagem para preparar o gravador e pressionar qualquer tecla, para iniciar o LOAD. Após o LOAD retornamos ao MENU.

A função 7 (SVT-7) permite que salvemos no cassete todas as 12 figuras ao mesmo tempo, sendo o procedimento similar ao da função 5 (SV-5).

A função 8 (LDT-8) permite que recuperemos as 12 figuras salvas pela função 7 (SVT-7), sendo o procedimento similar ao da função 6 (LD-6). Tanto a função 7 (SVT-7) como a função 8 (LDT-8) alteram a figura corrente para a figura 0.

A função 9 (COP-9) permite que copiemos a figura corrente em qualquer outra figura, bastando digitar 9 e em seguida o número da figura que irá receber a cópia da figura corrente. Em seguida retornamos ao MENU, com a figura corrente sendo mantida e a cópia sendo efetuada na figura especificada.

A função 0 (AN-0) permite que façamos uma simulação de animação, mostrando sequencialmente todas as 12 figuras na tela, até que seja digitado qualquer tecla. A figura corrente após o retorno ao MENU, poderá ser qualquer uma das 12 figuras.

Como podemos verificar, as 10 funções deste programa permitem uma grande versatilidade na edição e manipulação das figuras contidas neste livro, sendo de especial destaque a possibilidade de gravarmos figuras individuais no cassete, resultando em gravações compactas e com menor possibilidade de erros e

tempo de duração.

Para que possamos utilizar estas figuras gravadas no cassete em programas em Basic, foi desenvolvido o programa da listagem 3. Para entrarmos com este programa, devemos criar uma linha REM com pelo menos 128 caracteres e utilizaremos o programa da listagem 1, conforme explicado anteriormente. Depois do programa montado, este deverá ser gravado em cassete, antes de ser testado. Depois que fizermos a cópia poderemos testar com o comando:

```
RAND USR 16516
```

Com este comando, o RAMTOP será abaixado para 32500, a rotina será colocada a partir deste endereço e será executado um NEW, voltando o comando para o Basic. Para carregarmos uma só figura, devemos digitar os seguintes comandos:

```
POKE 32504,6
POKE 32557,192
POKE 32558,2
CLEAR
DIM A$(704)
RAND USR 32500
```

O CLEAR é necessário para assegurarmos que a matriz A$ seja a primeira a primeira na área de variáveis. Para carregarmos todas as 12 figuras devemos digitar os seguintes comandos:

```
POKE 32504,10
POKE 32557,0
POKE 32500,33
CLEAR
DIM F$(12,22,32)
RAND USR 32500
```

Depois de carregada a matriz A$, no caso de uma figura, ou a matriz F$, no caso de 12 figuras, é importante evitar o uso dos comandos RUN ou CLEAR, pois senão as figuras serão apagadas. Deve ser notado que, se a matrizes não estiverem corretamente dimensionadas (A$(704) ou F$(12,22,32)), poderemos perder o controle do computador.

LISTAGEM 1

```
  10 CLS
  20 PRINT "ENDERECO"
  30 INPUT I$
  40 IF LEN I$<>4 THEN GOTO 30
  50 LET I=(CODE I$-28)*4096+(CO
DE I$(2)-28)*256+(CODE I$(3)-28)
*16+CODE I$(4)-28
  60 SCROLL
  70 PRINT I$;"  ";
  80 INPUT X$
  90 IF X$="S" THEN STOP
 100 IF X$="R" THEN GOTO 10
 110 IF X$="E" THEN GOTO 260
 120 IF LEN X$<>16 THEN GOTO 90
 130 PRINT X$;"  ";
 140 LET A=0
 150 FOR J=1 TO 8
 160 LET Y$=X$( TO 2)
 170 LET X$=X$(3 TO )
 180 LET X=(CODE Y$-28)*16+CODE
Y$(2)-28
 190 POKE I,X
 200 LET I=I+1
 210 LET A=A+X
 220 NEXT J
 230 PRINT A
 240 GOSUB 1000
 250 GOTO 60
 260 PRINT AT 21,0;"
                    "
 270 LET I=I-8
 280 GOSUB 1000
 290 PRINT AT 21,0;
 300 GOTO 70
1000 LET I$=""
```

```
1010 LET X=I
1020 FOR K=1 TO 4
1030 LET X1=INT (X/16)
1040 LET X2=X-16*X1
1050 LET I$=CHR$ (X2+28)+I$
1060 LET X=X1
1080 NEXT K
1090 RETURN
```


LISTAGEM 2

```
408Z  76 76 AF 32 22 40 32 55  694
408A  45 CD 1D 41 54 5D 13 36  618
4092  00 01 FF 20 ED B0 CD 2A  948
409A  0A 01 00 16 CD F5 08 11  508
40A2  15 45 01 40 00 CD 6B 0B  478
40AA  01 01 17 CD F5 08 3A 55  626
40B2  45 C6 1C D7 CD 06 41 CD  991
40BA  F3 40 FE 76 C8 FE 1C CA  1363
40C2  77 41 FE 1D CA 4B 41 FE  1063
40CA  1E CA 1E 42 FE 1F CA AD  988
40D2  41 FE 20 CA 87 43 FE 21  1042
40DA  CA FD 43 FE 22 CA 5F 44  1175
40E2  FE 23 CA F1 43 FE 24 CA  1291
40EA  53 44 FE 25 CA 92 41 18  879
40F2  C6 CD BB 02 2C 20 FA CD  1123
40FA  BB 02 44 4D 2C 28 F8 CD  871
4102  BD 07 7E C9 CD 1D 41 ED  1059
410A  5B 0C 40 13 1A FE 76 28  624
4112  07 7E 12 0B 78 B1 C8 23  694
411A  13 18 F1 21 00 5E 01 C0  604
4122  02 3A 55 45 3C 3D C8 09  544
412A  18 FB 21 47 41 AF 4E 23  732
4132  46 EB ED 42 38 03 3C 18  751
413A  F9 09 C6 1C D7 EB 79 FE  1309
4142  01 C8 23 18 E8 0A 00 01  503
414A  00 CD 51 41 C3 98 40 CD  967
4152  2A 0A CD 06 41 01 00 16  351
415A  CD F5 08 11 D8 41 01 18  781
4162  00 CD 6B 0B CD F3 40 FE  1089
416A  1C 38 F9 FE 28 30 F5 D6  1134
4172  1C 32 55 45 C9 CD 2A 0A  690
417A  CD 06 41 3A 55 45 3C FE  802
4182  0C 38 01 AF 32 55 45 CD  653
418A  BB 02 2C C2 98 40 18 E8  899
4192  3A 55 45 F5 CD 51 41 CD  1013
419A  1D 41 E3 7C 32 55 45 CD  854
```

41A2	1D	41	D1	01	C0	02	ED	B0	911
41AA	C3	98	40	CD	2A	0A	CD	06	879
41B2	41	01	00	17	CD	F5	08	11	564
41BA	F0	41	01	1B	00	CD	6B	0B	656
41C2	CD	F3	40	FE	38	C2	98	40	1232
41CA	CD	1D	41	54	5D	13	0B	36	560
41D2	00	ED	B0	C3	98	40	36	3A	936
41DA	26	31	00	34	00	33	3A	32	298
41E2	2A	37	34	00	29	26	00	2B	271
41EA	2E	2C	3A	37	26	0E	36	3A	367
41F2	2A	37	00	37	2A	26	31	32	331
41FA	2A	33	39	2A	00	26	35	26	321
4202	2C	26	37	00	10	38	18	33	284
420A	11	31	2E	33	2D	26	0E	00	260
4212	00	00	00	00	00	28	34	31	141
421A	3A	33	26	0E	CD	2A	0A	CD	623
4222	06	41	AF	32	74	43	32	75	646
422A	43	01	00	17	CD	F5	08	11	566
4232	0B	42	01	13	00	CD	6B	0B	420
423A	3E	2C	32	76	43	01	06	17	371
4242	CD	F5	08	3A	75	43	5F	16	817
424A	00	CD	2C	41	01	13	17	CD	562
4252	F5	08	3A	74	43	5F	16	00	611
425A	CD	2C	41	CD	BB	02	2C	28	792
4262	08	01	00	10	0B	78	B1	20	365
426A	FB	3A	75	43	4F	06	00	60	674
4272	69	29	29	29	29	29	09	3A	377
427A	74	43	4F	06	00	09	EB	2A	554
4282	0C	40	23	19	56	3A	76	43	465
428A	77	D5	E5	CD	BB	02	44	4D	1100
4292	3E	FF	2C	28	04	CD	BD	07	806
429A	7E	E1	D1	72	06	00	10	FE	950
42A2	FE	FF	28	E0	FE	76	CA	59	1436
42AA	43	FE	70	28	5B	FE	71	28	971
42B2	4A	FE	72	28	3D	FE	73	28	952
42BA	30	FE	74	CA	77	43	FE	75	1177
42C2	CA	7F	43	FE	DA	CA	83	43	1268
42CA	FE	40	30	B8	E5	4F	3A	76	1034

42D2	43	FE	2C	20	07	21	D5	44	718
42DA	06	00	09	4E	FE	2E	20	02	427
42E2	CB	F9	79	E1	FE	FF	28	9C	1503
42EA	77	3A	74	43	3C	32	74	43	653
42F2	18	20	3A	74	43	3D	32	74	524
42FA	43	18	17	3A	75	43	FE	15	631
4302	28	10	3C	32	75	43	18	0A	384
430A	3A	75	43	A7	28	04	3D	32	564
4312	75	43	3A	74	43	FE	20	20	743
431A	0D	AF	32	74	43	3A	75	43	663
4322	3C	32	75	43	18	10	FE	FF	843
432A	20	0C	3E	1F	32	74	43	3A	428
4332	75	43	3D	32	75	43	3A	75	654
433A	43	FE	16	20	0C	3E	15	32	520
4342	75	43	3E	1F	32	74	43	18	534
434A	0B	FE	FF	20	07	AF	32	75	901
4352	43	32	74	43	C3	3F	42	CD	829
435A	1D	41	EB	2A	0C	40	23	7E	608
4362	FE	76	20	03	23	18	F8	12	732
436A	23	13	0B	78	B1	20	F0	C3	829
4372	98	40	00	00	00	3E	2C	32	372
437A	76	43	C3	86	42	3E	33	18	717
4382	F6	3E	2E	18	F2	CD	2A	0A	877
438A	CD	1D	41	EB	2A	0C	40	23	687
4392	1A	C5	E5	06	40	0E	00	21	569
439A	D5	44	BE	28	0B	0C	23	10	585
43A2	F9	FE	80	30	04	AF	18	01	883
43AA	79	E1	C1	F5	7E	FE	76	20	1314
43B2	01	23	F1	77	23	13	0B	78	581
43BA	B1	20	D5	01	00	16	CD	F5	895
43C2	08	11	3A	44	01	19	00	CD	382
43CA	6B	0B	CD	F3	40	C3	98	40	1041
43D2	CD	2A	0A	CD	06	41	01	00	534
43DA	16	CD	F5	08	11	25	44	01	603
43E2	2E	00	CD	6B	0B	CD	F3	40	881
43EA	CD	2A	0A	CD	1D	41	C9	AF	932

LISTAGEM 3

43F2	32	55	45	CD	D2	43	01	00	687
43FA	21	18	03	CD	D2	43	C5	E5	968
4402	CD	E7	02	21	A0	0F	10	FE	916
440A	2B	7C	B5	20	F9	CD	1E	03	867
4412	E1	C1	C5	CD	1E	03	C1	23	1081
441A	0B	78	B1	20	F5	CD	07	02	799
4422	C3	98	40	35	37	2A	35	26	652
442A	37	2A	00	34	00	2C	37	26	286
4432	3B	26	29	34	37	00	2A	00	287
443A	35	37	2A	38	38	2E	34	33	411
4442	2A	00	00	36	3A	26	31	36	295
444A	3A	2A	37	00	39	2A	28	31	343
4452	26	AF	32	55	45	CD	D2	43	899
445A	01	00	21	18	03	CD	D2	43	543
4462	22	D1	44	ED	43	D3	44	CD	1099
446A	E7	02	AF	32	21	40	CD	75	877
4472	44	18	FB	0E	01	06	00	3E	426
447A	7F	DB	FE	D3	FF	1F	30	4D	1222
4482	17	17	38	2C	10	F1	F1	3A	702
448A	21	40	CB	47	C2	E5	03	CD	1002
4492	75	44	3A	21	40	CB	C7	ED	979
449A	4B	D3	44	2A	D1	44	C5	CD	1075
44A2	75	44	71	C1	23	0B	78	B1	834
44AA	20	F4	CD	07	02	C3	98	40	901
44B2	D5	1E	94	06	1A	1D	DB	FE	925
44BA	17	CB	7B	7B	38	F5	10	F5	1034
44C2	D1	20	04	FE	56	30	AE	3F	870
44CA	CB	11	30	A9	C9	CF	0C	00	857
44D2	5E	00	21	00	FF	FF	FF	FF	1147
44DA	FF	FF	FF	FF	FF	FF	FF	FF	2040
44E2	FF	FF	FF	FF	FF	FF	FF	FF	2040
44EA	FF	FF	FF	FF	FF	FF	FF	FF	2040
44F2	01	02	87	04	05	83	03	85	414
44FA	FF	88	FF	FF	09	07	8A	89	1192
4502	08	FF	FF	FF	FF	FF	FF	FF	1793
450A	80	81	84	0A	06	FF	FF	82	1045

4512	FF	86	FF	2B	2E	2C	00	3B	836
451A	2A	37	16	1D	00	26	31	39	292
4522	16	1E	00	26	35	2C	16	1F	240
452A	00	31	38	39	16	20	00	38	272
4532	3B	16	21	00	00	00	00	31	163
453A	29	00	16	22	00	38	3B	39	269
4542	16	23	00	31	29	39	16	24	262
454A	00	28	34	35	16	25	00	26	242
4552	33	16	1C	0B	C3	3C	7A	00	489
4082	76	76	21	F4	7E	22	04	40	741
408A	EB	21	96	40	01	6B	00	ED	827
4092	B0	C3	C3	03	2A	10	40	01	692
409A	06	00	09	EB	CD	E7	02	AF	863
40A2	32	21	40	CD	08	7F	18	FB	762
40AA	0E	01	06	00	3E	7F	DB	FE	683
40B2	D3	FF	1F	30	48	17	17	38	719
40BA	27	10	F1	F1	3A	21	40	CB	895
40C2	47	C2	E5	03	CD	08	7F	3A	895
40CA	21	40	CB	C7	01	C0	02	EB	929
40D2	C5	CD	08	7F	71	C1	23	0B	889
40DA	78	B1	20	F4	CD	07	02	C9	988
40E2	D5	1E	94	06	1A	1D	DB	FE	925
40EA	17	CB	7B	7B	38	F5	10	F5	1034
40F2	D1	20	04	FE	56	30	B3	3F	875
40FA	CB	11	30	AE	C9	CF	0C	00	862

USANDO AS FIGURAS COM LINGUAGEM DE MÁQUINA

Os melhores efeitos no vídeo só podem ser obtidos a partir de programas diretamente na linguagem da máquina (microprocessadora) capazes de produzir a velocidade que programas interpretados em Basic não dispõem.

O programa ENTER GRAPHICS, em código, é o melhor começo e uma ótima ferramenta. Além das facilidades de edição possui rotinas adicionais, como para a gravação e carga individual das figuras, completando-se a sua versatilidade com as rotinas das listagens.

Por último, relembro que, pará utilizar as rotinas propostas a seguir, bem como o próprio programa ENTER GRAPHICS em código, não é necessário conhecimento de programação Assembly Z-80.

Para mostrarmos algumas das aplicações das figuras dentro de programas em Basic, incluimos a listagem 4, que consiste em 3 rotinas escritas em linguagem de máquina, que nos possibilitam rodar o vídeo para a esquerda, rodar o vídeo para a direita e uma simulação de animação, mostrando sequencialmente as 12 figuras no vídeo.

Para colocarmos este programa no computador devemos criar uma linha REM com pelo menos 160 caracteres e, com o auxílio do monitor da listagem 1, digitamos o programa.Antes de testar devemos fazer uma cópia.

Para rodarmos a primeira rotina, devemos colocar o número

da linha em que começaremos a rodar o vídeo para a esquerda na posição 16516. E o número da última linha a ser rodada na posição 16517. Os números de linha permitidos variam de 0 a 23, não sendo checado se estes números estão dentro da faixa permitida e se o número inicial é menor que o inicial. Depois de "POKADOS" os valores, chamados a rotina através de RAND USR 16518.

Por exemplo, se colocarmos "0" em 16516 e "23" em 15517, quando chamarmos a rotina através de RAND USR 16518, todas as 24 linhas do vídeo serão rodadas uma posição para a esquerda, sendo o conteúdo da margem esquerda copiado na margem direita.

Mas se colocarmos "9" em 16516 e "12" em 16517, somente as linhas 9, 10, 11 e 12 serão rodadas, sendo que o resto do vídeo permanece inalterado.

Para a segunda rotina, rodar o vídeo para a direita, o procedimento é idêntico somente variando o endereção de chamada, que é executada pelo comando RAND USR 16567.

Para utilizarmos a terceira rotina, devemos primeiro ter dimensionado uma matriz F$(12,22,32) e esta matriz deverá ser a primeira na área de variáveis. A chamada será atravós de RAND USR 16622. E, a cada chamada, será copiado no vídeo uma figura de numeração maior que a anterior. Se quiser começar por uma figura específica, basta colocar na posição o seu número, na faixa de 1 a 12.

Para a quarta rotina, inversor de vídeo, bastará o comando RAND USR 16671 e, o vídeo será invertido.

Para a quinta rotina, simulação de tiro, devemos colocar o número da coluna na posição 16716 e o número da linha na posição 16717, pelo comando POKE. A rotina será chamada pelo comando RAND USR 16692.

Na primeira chamada o caractere que estiver na posição indicada será copiado na linha superior, e a posição original será apagada. Chamando-se esta rotina repetidas vezes o caractere subirá até sumir no topo do vídeo.

LISTAGEM 4

4082	76	76	00	17	3A	84	40	3C	573
408A	01	21	00	2A	0C	40	23	3D	248
4092	28	03	09	18	FA	54	5D	23	538
409A	ED	4B	84	40	78	91	3C	47	904
40A2	1A	F5	7E	FE	76	28	05	12	832
40AA	23	13	18	F6	F1	12	23	23	653
40B2	13	13	10	EC	C9	2A	0C	40	609
40BA	01	17	03	09	3A	85	40	47	362
40C2	3E	18	90	01	DF	FF	3D	28	810
40CA	03	09	18	FA	54	5D	2B	ED	743
40D2	4B	84	40	78	91	3C	47	1A	693
40DA	F5	7E	FE	76	28	05	12	2B	849
40E2	1B	18	F6	F1	12	2B	2B	1B	669
40EA	1B	10	EC	C9	2A	10	40	01	603
40F2	0A	00	09	3A	1E	41	FE	0C	438
40FA	38	01	AF	3C	32	1E	41	01	438
4102	C0	02	3D	28	03	09	18	FA	581
410A	ED	5B	0C	40	13	1A	FE	76	821
4112	28	07	7E	12	0B	78	B1	C8	699
411A	23	13	18	F1	00	2A	0C	40	437
4122	23	06	16	7E	FE	76	28	06	607
412A	EE	80	77	23	18	F5	23	10	840
4132	F2	C9	ED	4B	4C	41	CD	4E	1179
413A	41	56	36	00	05	78	FE	FF	839
4142	C8	ED	43	4C	41	CD	4E	41	993
414A	72	C9	00	00	C5	F5	2A	0C	811
4152	40	23	78	3C	06	00	09	01	295
415A	21	00	3D	28	03	09	18	FA	420
4162	F1	C1	C9	00	00	00	00	00	635

EXEMPLO A — ANIMAÇÃO

O próprio programa Enter Graphics, versão em código, possui uma rotina para animação. Opção **0** do Menu. Um bom exemplo e fácil de entrar é com os bonecos da página 91.

Usando-se apenas os quatro da primeira linha e o primeiro da esquerda da segunda linha, entre-os no Enter Graphics como figuras **0**, 1, 2, 3 e 4, sempre na mesma posição. A seguir usando a opção 9 do Menu, que copia a figura corrente em outra figura, complete as demais usando as anteriores como a seguir: 5 e 6 igual a 4, 7 igual a 3, 8 igual a 2, 9 igual a 1 e AeB igual a **0**. As figuras iguais em seguida, 4, 5 e 6 e A, B e **0** visam uma pequena parada nestas posições.

Teste a animação pela opção **0**.

Mesmo que você use figuras com a tela completa a velocidade será igual a este exemplo, pois a opção **0** AN copia a tela completa.

EXEMPLO B: MOVIMENTOS DIVERSOS

Este outro exemplo, toma "uma determinada imagem impressa, e desloca-a através do vídeo, quer seja para cima, para baixo ou para os lados, o vídeo todo, uma parte, ou mais de uma parte.

Comece com a listagem 1 das páginas 59 e 60, carregando os códigos da listagem 4, página 68, sem o programa Enter Graphics.

Esta rotina, fará com que um tanque de guerra fique passando pela tela da esquerda para a direita. A qualquer tecla pressionada (exceto Break) um míssel é disparado. Se o tanque for atingido é dada a explosão pela inversão do vídeo 4 vezes consecutivas. Ao final de 10 tiros é dado o número de acertos e erros com a inversão do vídeo. A dectetação do ponto de encontro ou explosão é dado pela leitura da variável Current Caracter, nos endereços 16398 e 16399 a linha nº 1605.

De qualquer forma, isto é apenas um exemplo de o que se pode fazer "com os desenhos e as rotinas da listagem 4". Espero que você deseje, não tudo mastigado, mas os subsídios para elaborar seus próprios programas animados.

A figura das linhas 1100 a 1120 é a primeira da página 98,

```
1100 PRINT "▀▀▀▀▀▀▀▀▀▀▀▀"
1110 PRINT "▀▀▀▀▀▀▀▀▀▀▀▀"
1120 PRINT "▀▀▀▀▀▀▀▀▀▀▀▀"
1121 LET E=0
1122 POKE 16516,0
1123 POKE 16517,2
1124 LET A=0
1125 LET S=0
1130 LET T=0
1135 RAND
1136 LET R=INT (RND*10)
1140 PRINT AT 21,R;"+"
1142 POKE 16716,R
1144 POKE 16717,21
1250 RAND USR 16518
1260 IF INKEY$<>"" THEN LET S=1
1280 IF PEEK 16717=3 THEN GOTO 1
600
1290 IF S=1 THEN GOSUB 1400
1300 GOTO 1250
1330 GOTO 1135
1400 RAND USR 16692
1420 RETURN
1500 FOR Z=1 TO 8
1510 RAND USR 16671
1520 NEXT Z
1522 LET A=A+1
1525 IF T=19 THEN GOTO 1125
1530 RETURN
1600 PRINT AT 3,R;" ";
1605 LET D=PEEK ((PEEK 16398+256
*PEEK 16399)-34)
1610 IF D<>0 THEN GOSUB 1500
1620 LET E=E+1
1625 IF E=10 THEN GOTO 1650
1630 GOTO 1125
1650 PRINT AT 10,5;"FORAM DEZ CH
ANCES"
1660 PRINT AT 12,2;"...E VOCE AC
ERTOU ";A
1670 RAND USR 16671
```


PARTE IV

OS DUZENTOS DESENHOS

OS 200 DESENHOS

Entre as figuras fornecidas neste capítulo espero que você encontre a que deseja. Em todo caso, se não encontrar, certamente nos outros capítulos você obterá um meio de videomicrar a sua figura.

Quanto às figuras fornecidas, você observará que na mesma página, a mesma figura é representada por letras e/ou números das teclas dos respectivos símbolos gráficos.

A razão de utilizarmos esta codificação intrínseca ao teclado Sinclair, é simples de compreender. Vejamos a espaçonave abaixo:

É formada por quais caracteres gráficos?

Seria ou ?

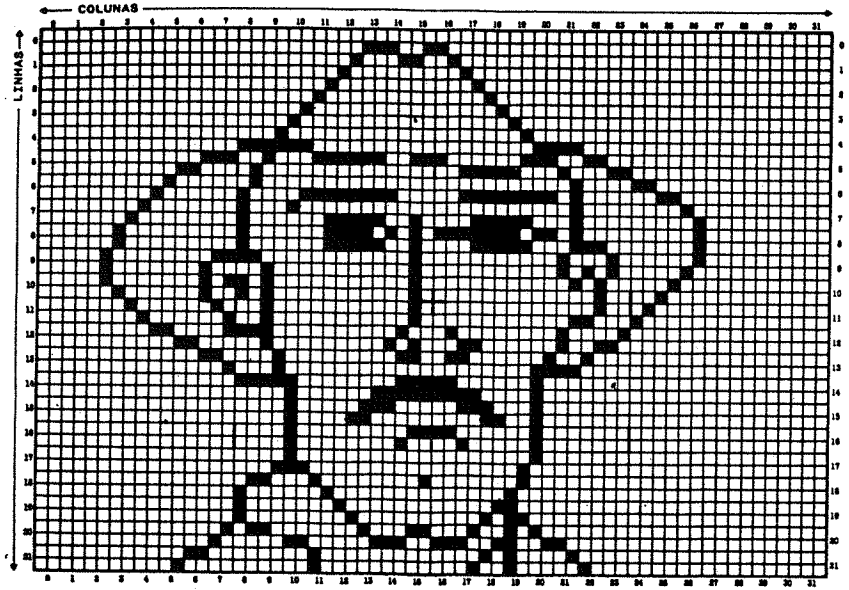
Quadriculando-se a imagem é possivel a identificação dos caracteres, mas daria para lê-los se fossem soletrados, ditando-se e identificando-os rapidamente no teclado?

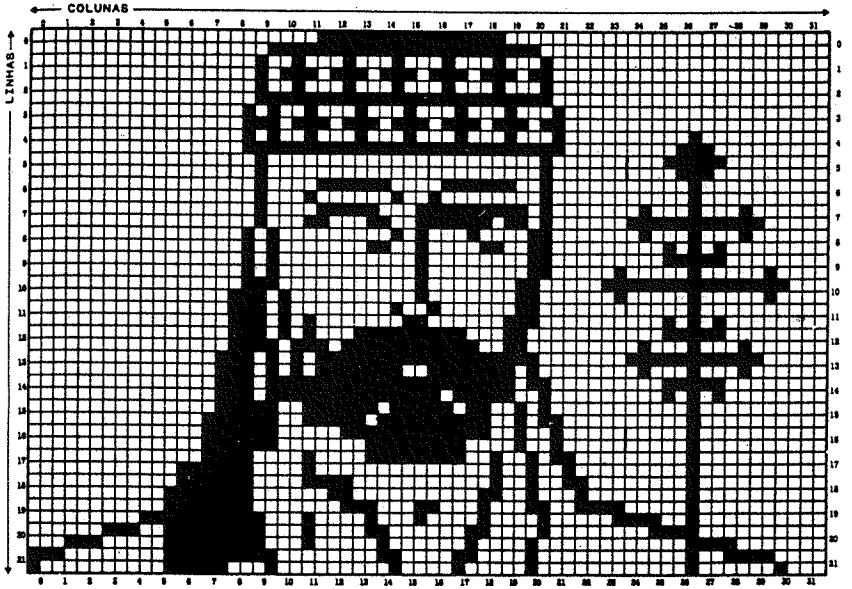
Por exemplo: o que você prefere, ler ou digitar:

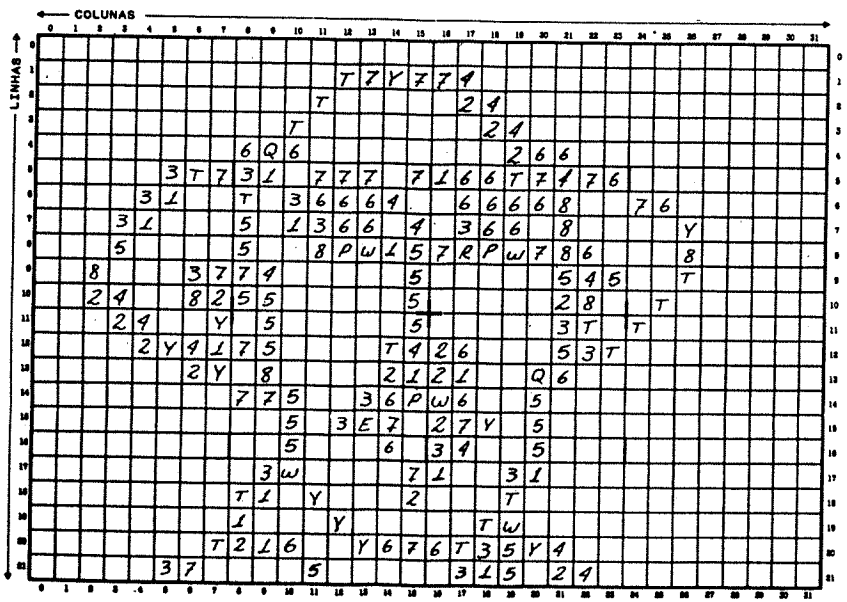
" " ou "1234567 "

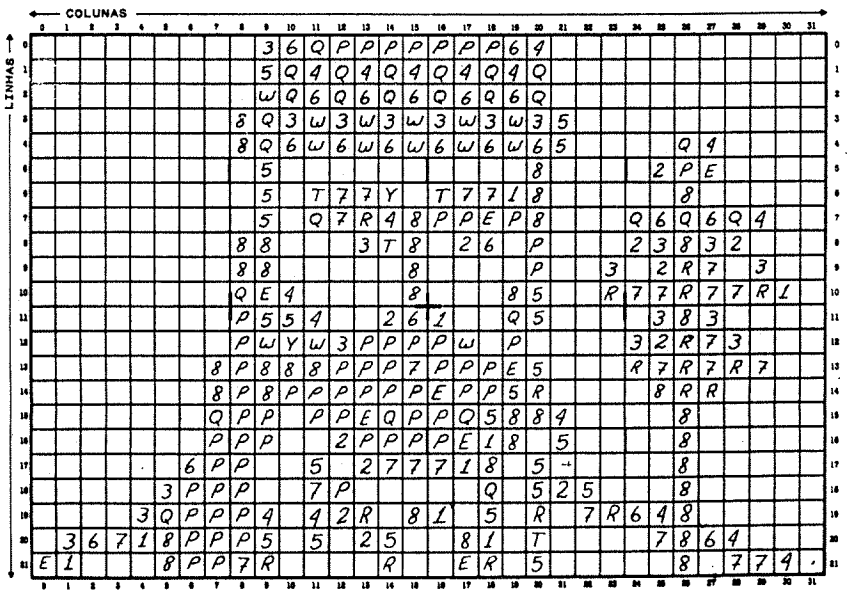
Antes de tentar o seu modo tente o nosso. Veja como entrar e usar figuras a partir do programa ENTER GRAPHICS, nas versões Basic e Linguagem de Máquina.

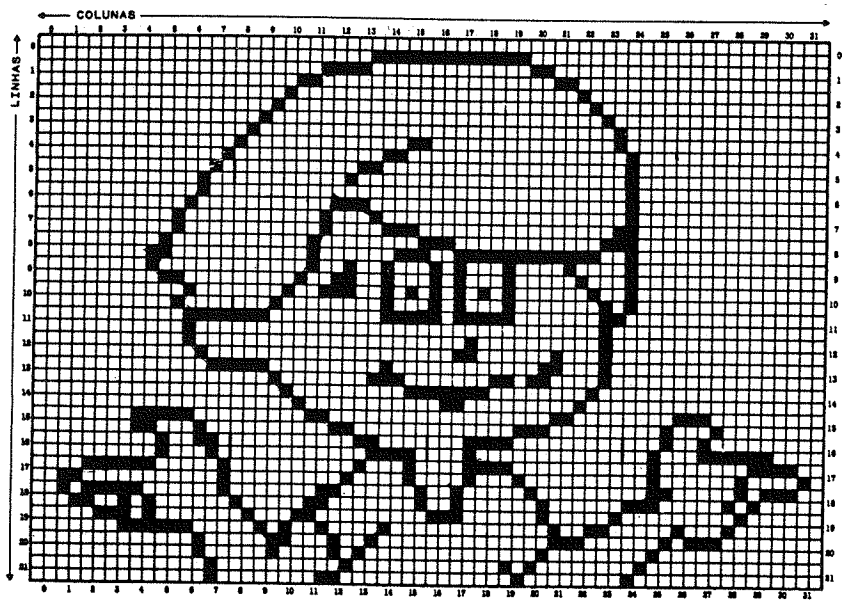
Observação: A letra P, nas listagens a seguir, podendo ser digitada nos programas Enter Graphics, significa um inverse space.

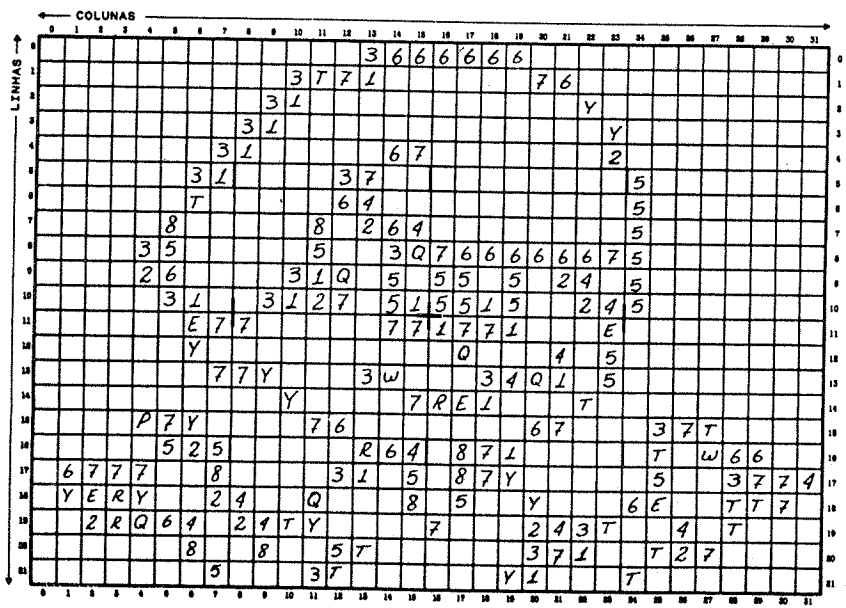

COLUNAS
LINHAS

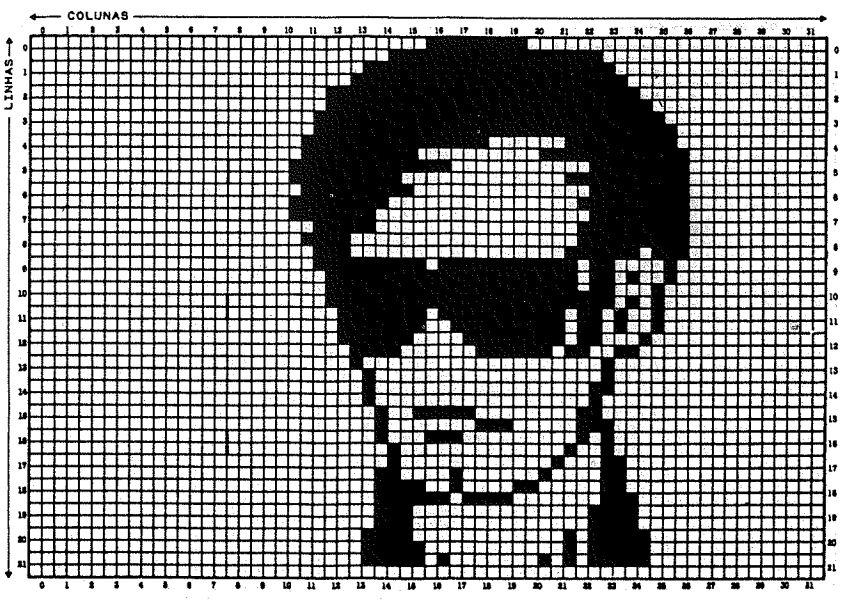

COLUNAS
LINHAS

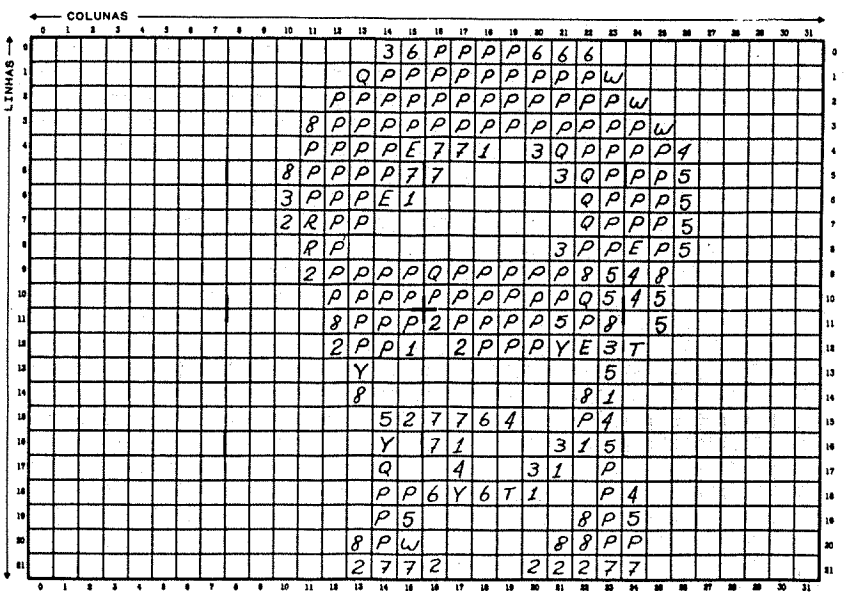

COLUNAS
LINHAS

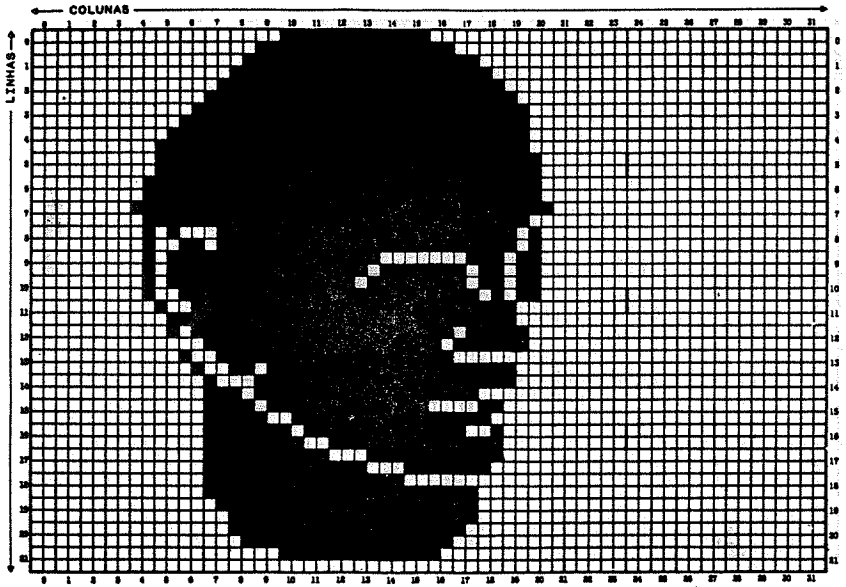

COLUNAS
LINHAS

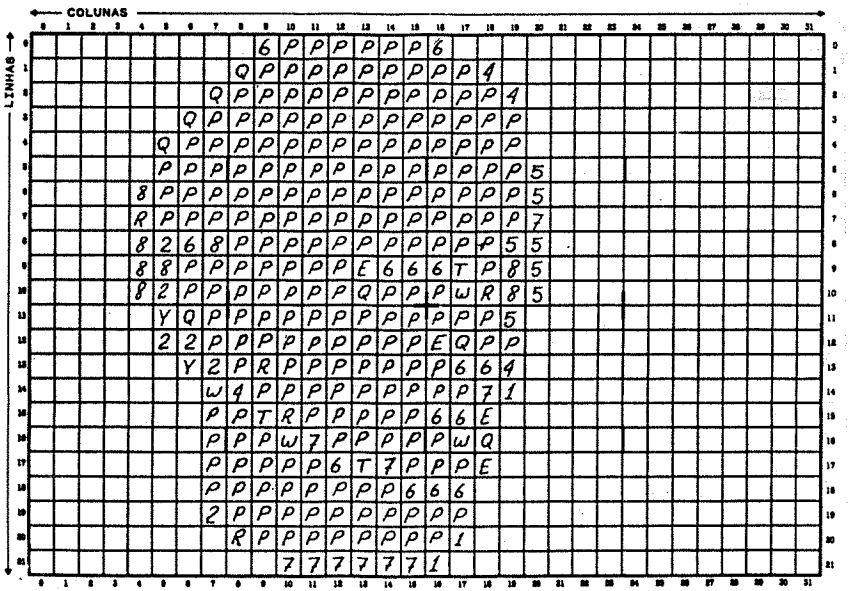

COLUNAS
LINHAS

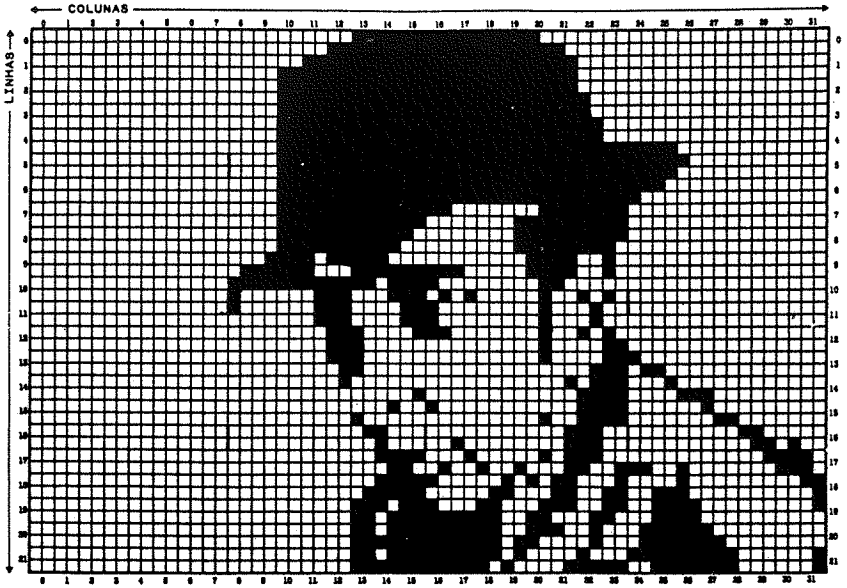

COLUNAS
LINHAS

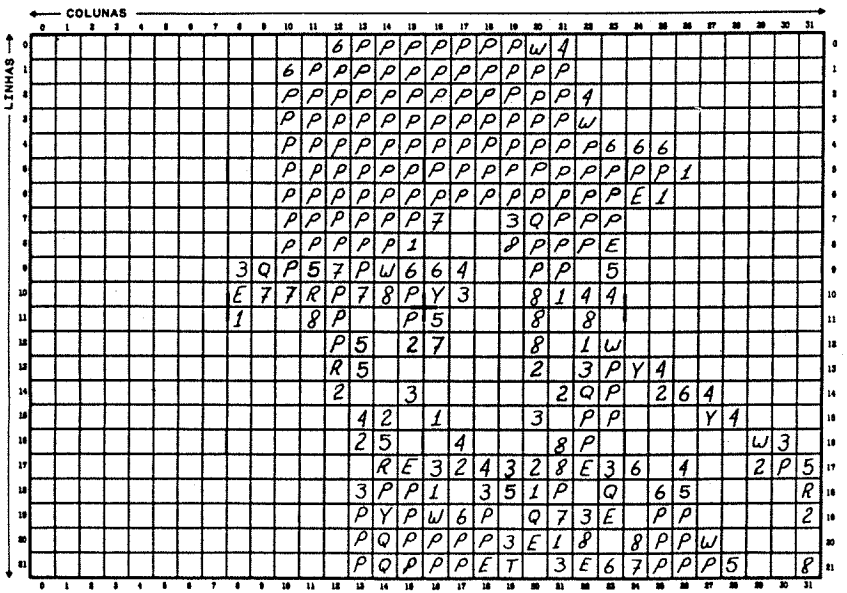

COLUNAS
LINHAS

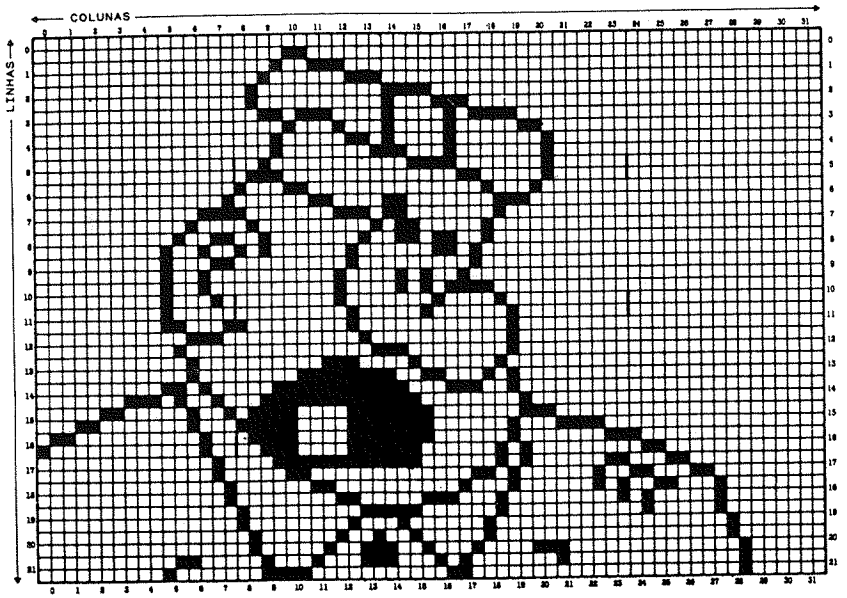

COLUNAS
LINHAS

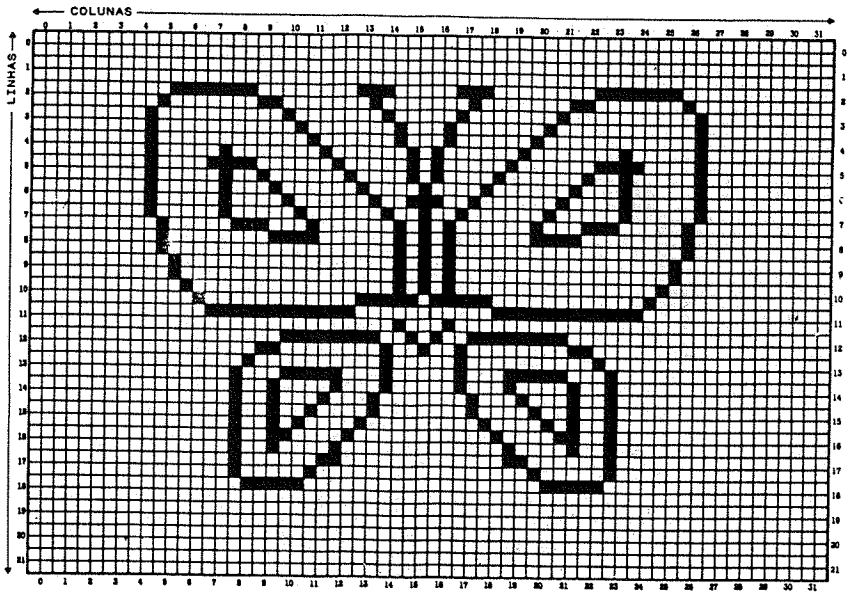

COLUNAS
LINHAS

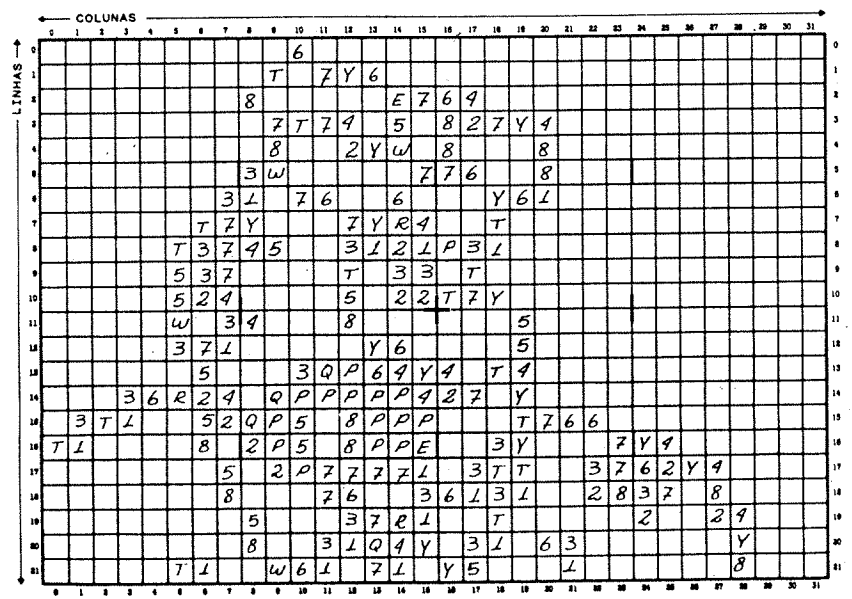

COLUNAS
LINHAS


COLUNAS
LINHAS

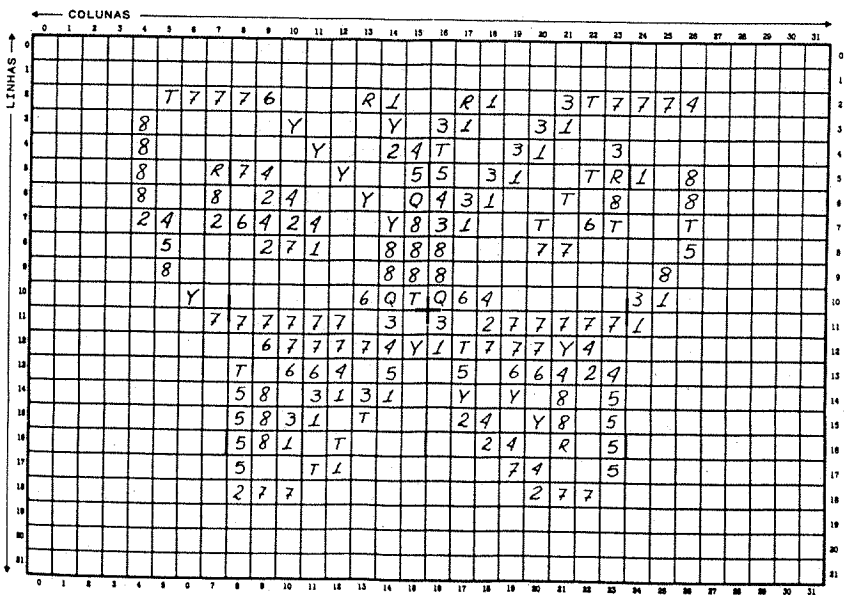

COLUNAS
LINHAS


COLUNAS
LINHAS

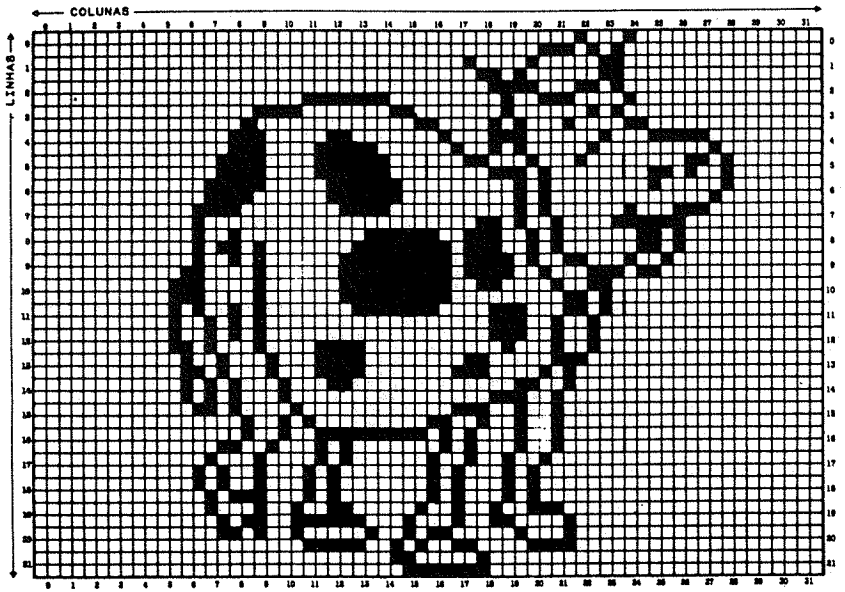

COLUNAS
LINHAS

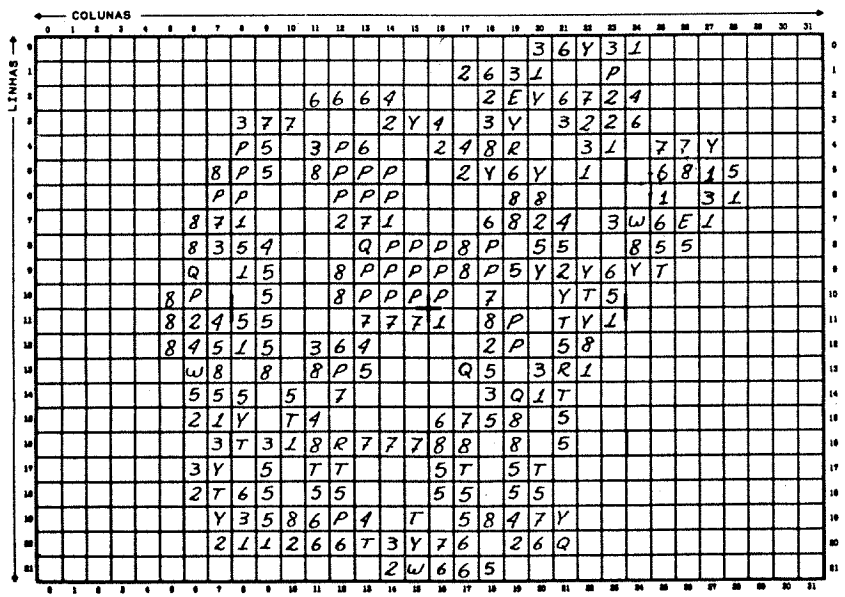

COLUNAS
LINHAS

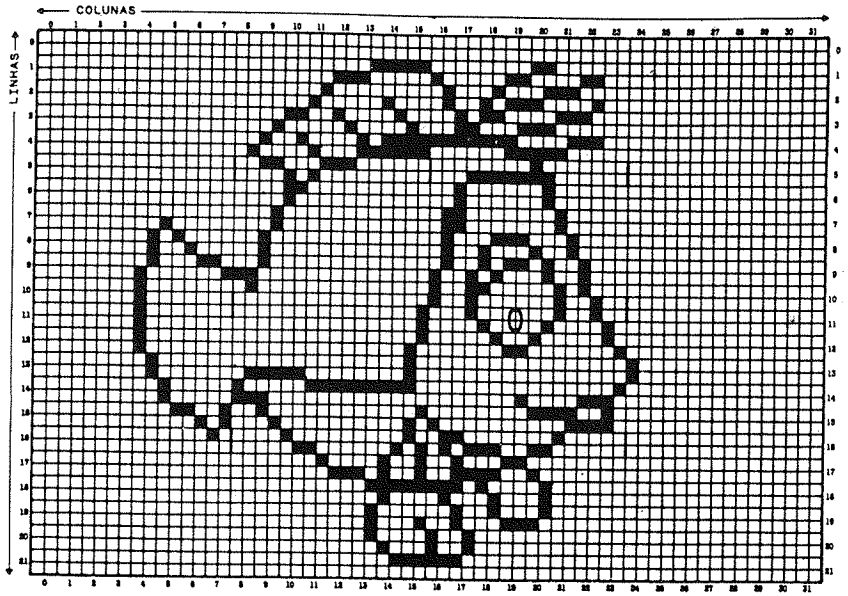
80

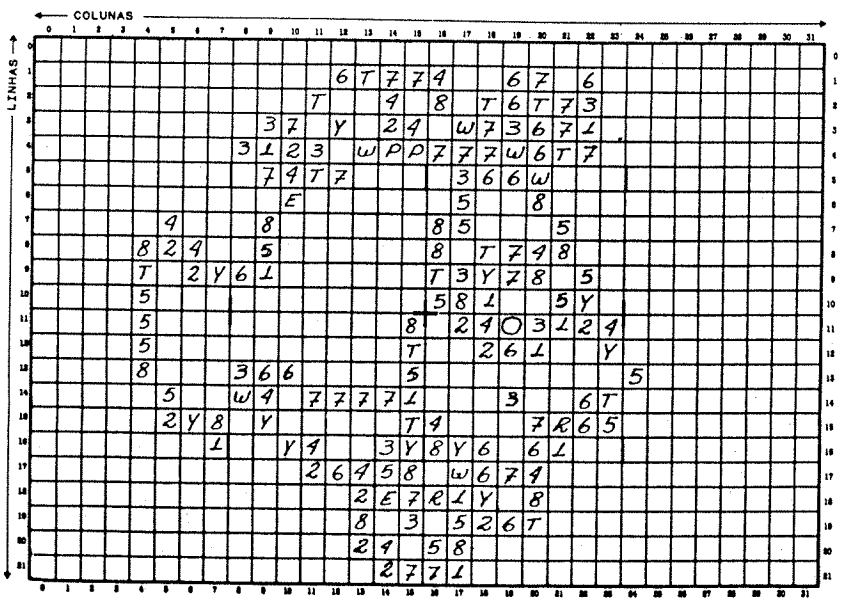

COLUNAS
LINHAS

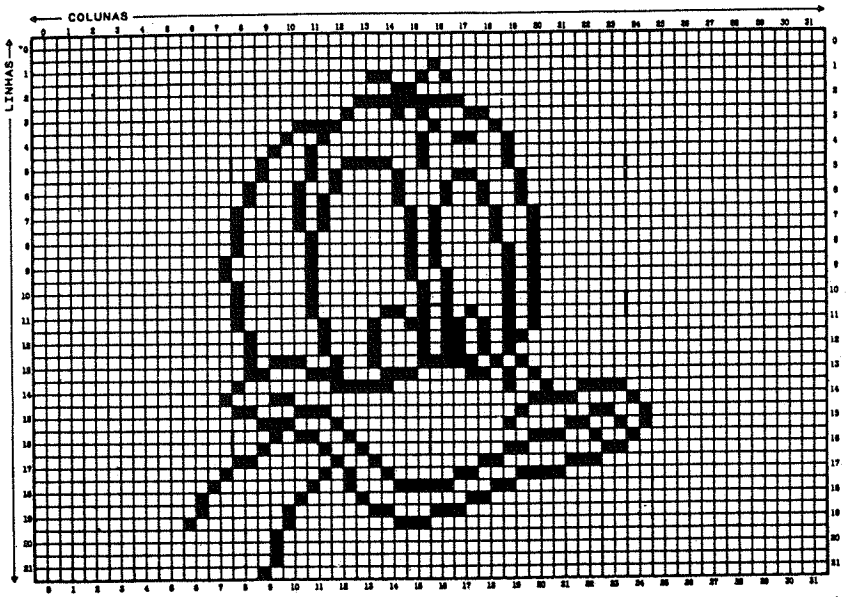
81

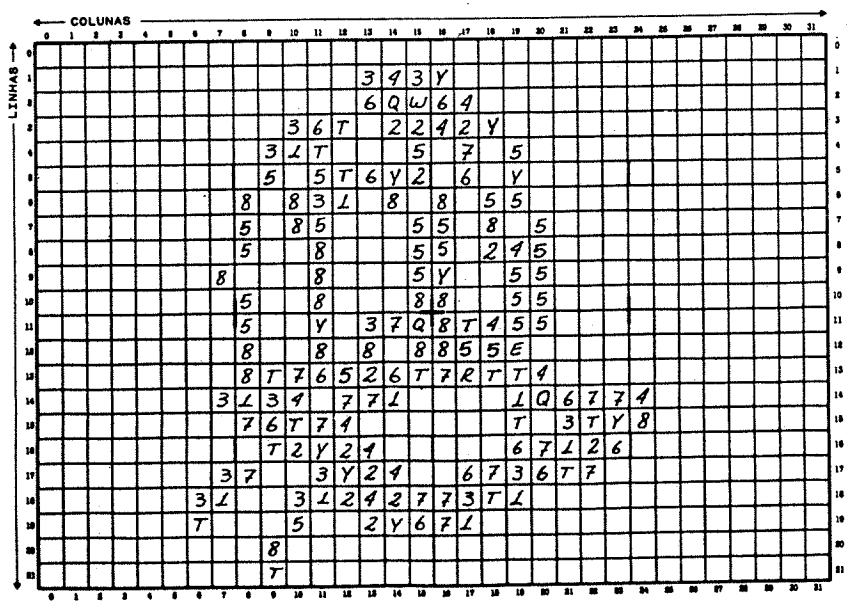

COLUNAS
LINHAS

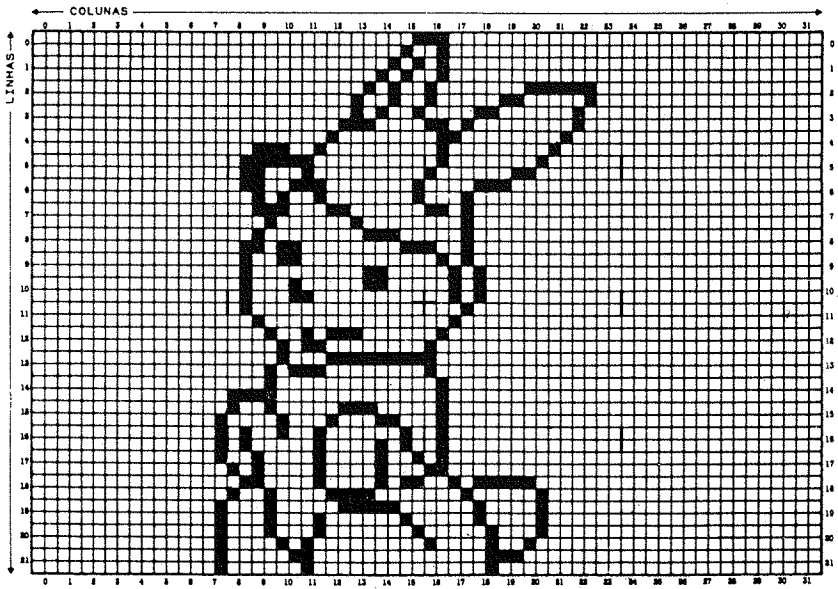

COLUNAS
LINHAS

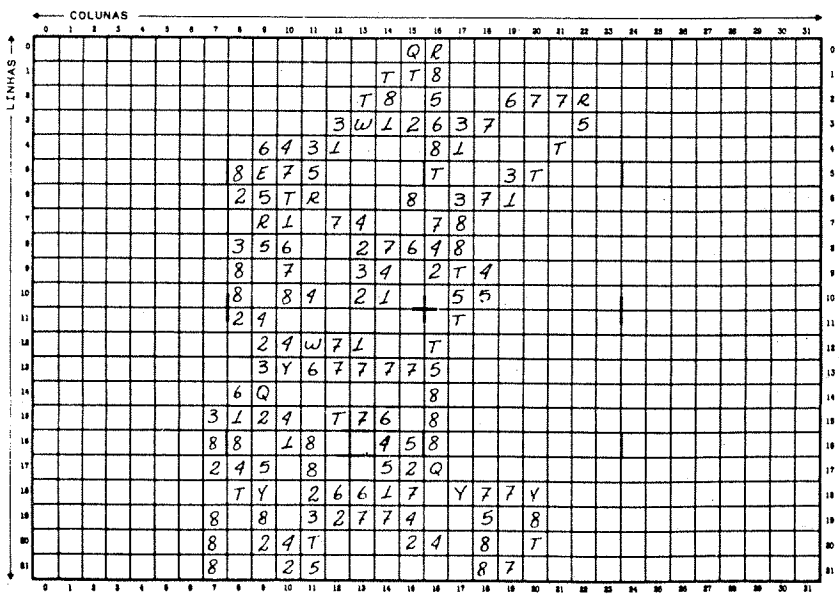

COLUNAS
LINHAS

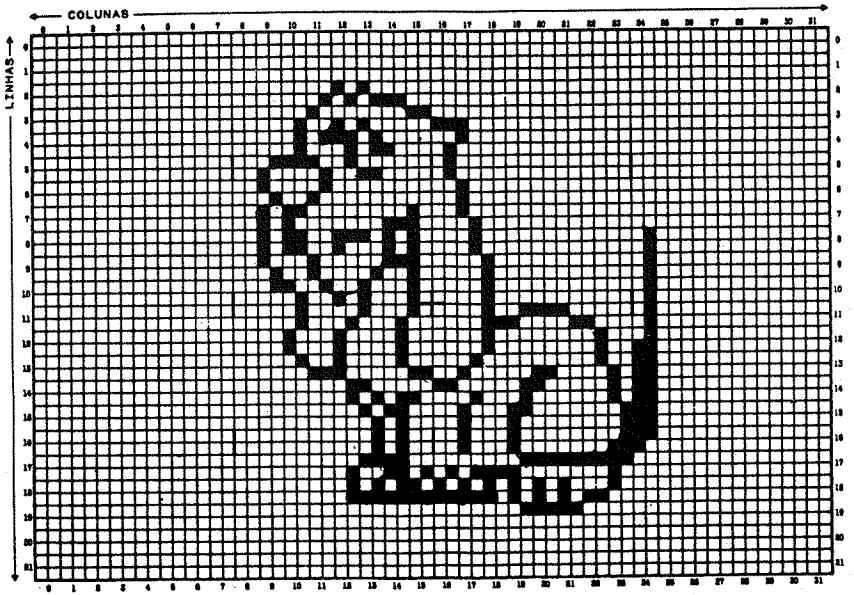

COLUNAS
LINHAS

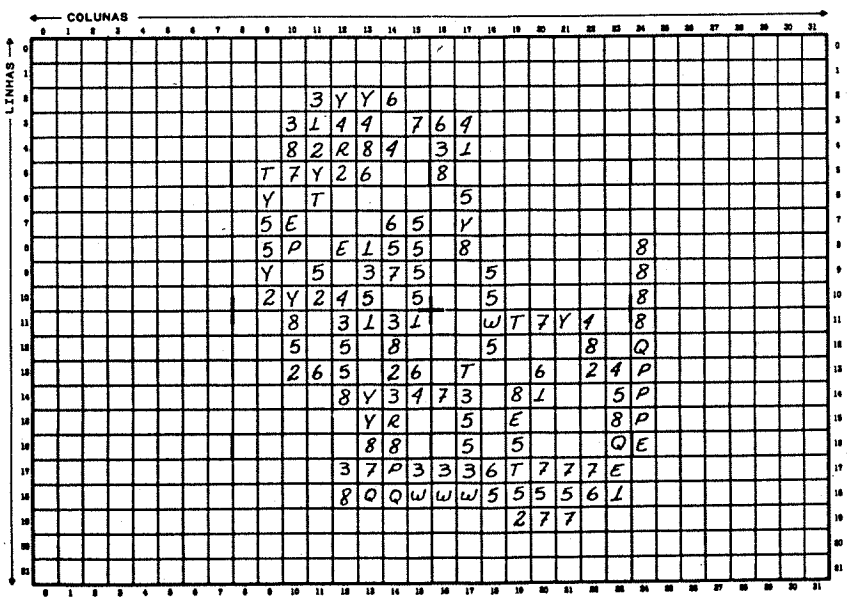

COLUNAS
LINHAS

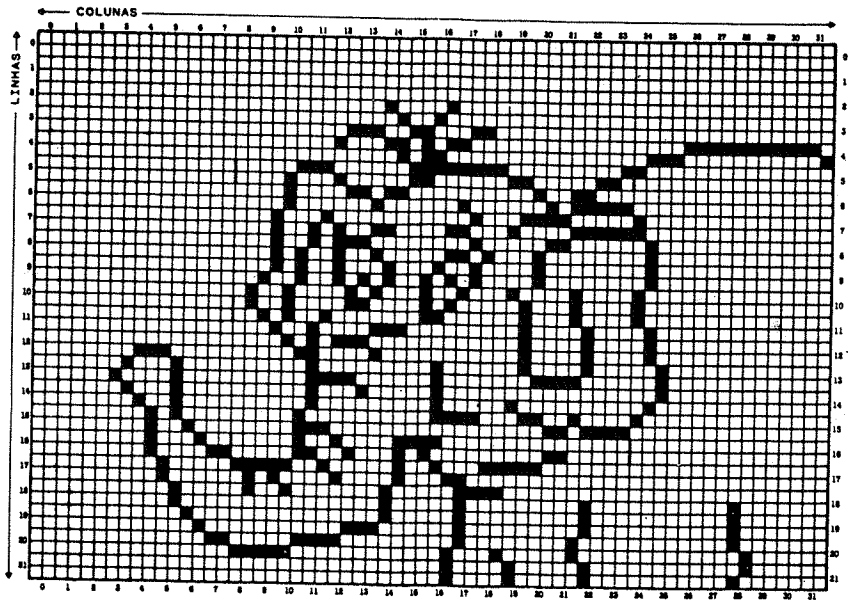

COLUNAS
LINHAS

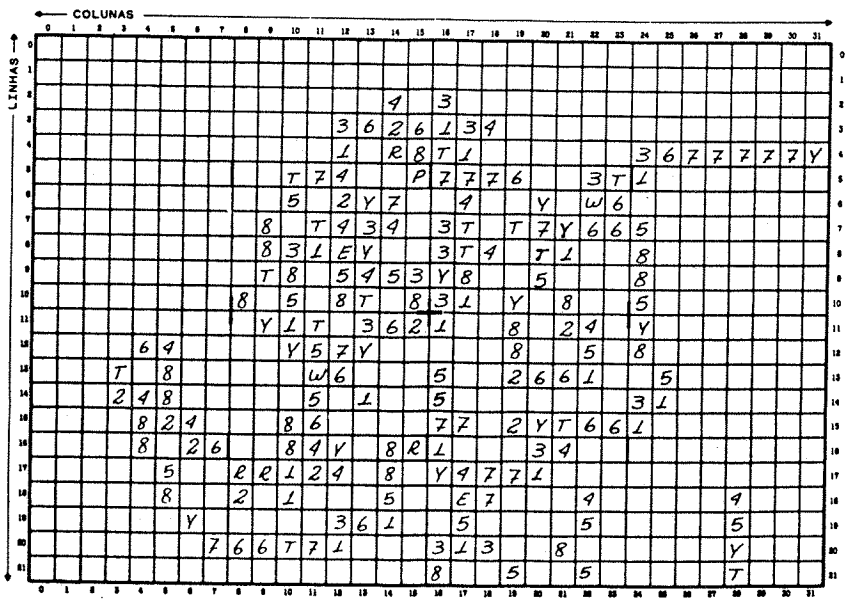

COLUNAS
LINHAS

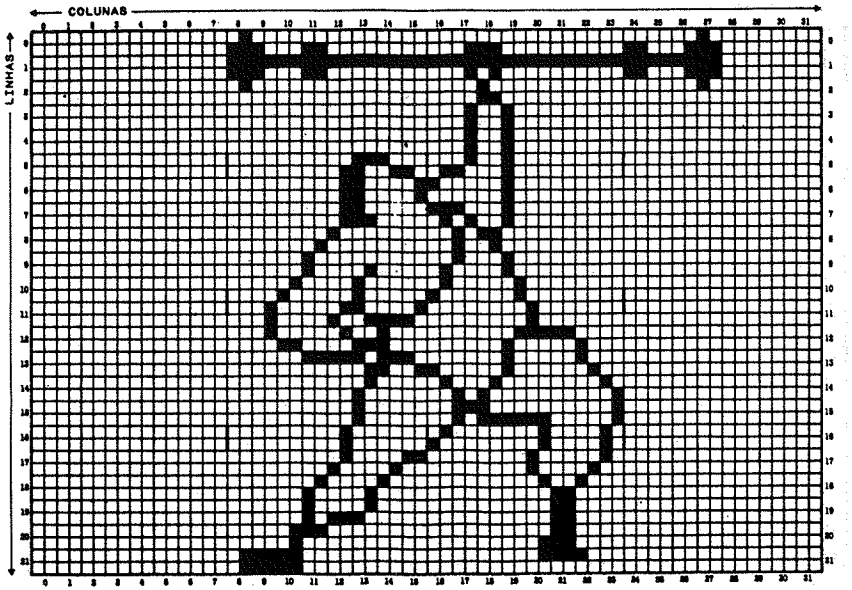

COLUNAS
LINHAS

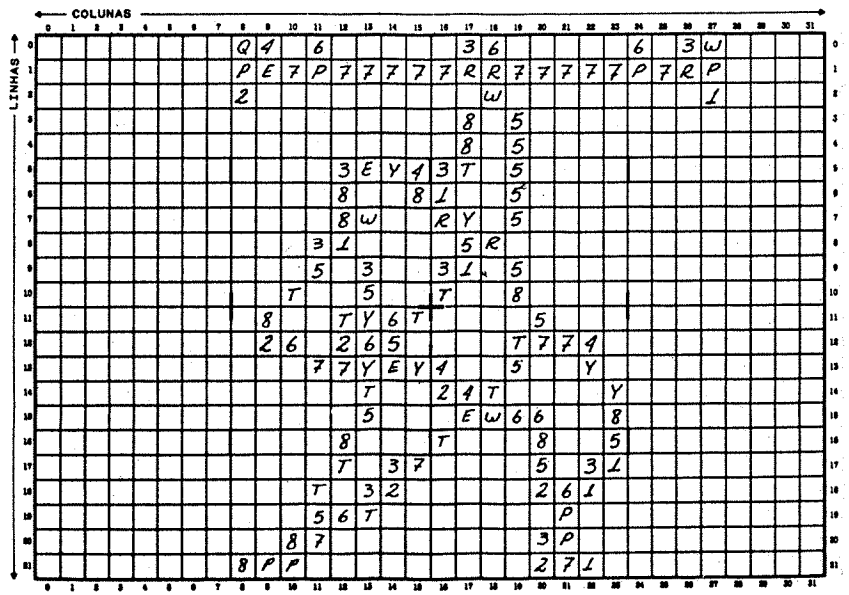

COLUNAS
LINHAS

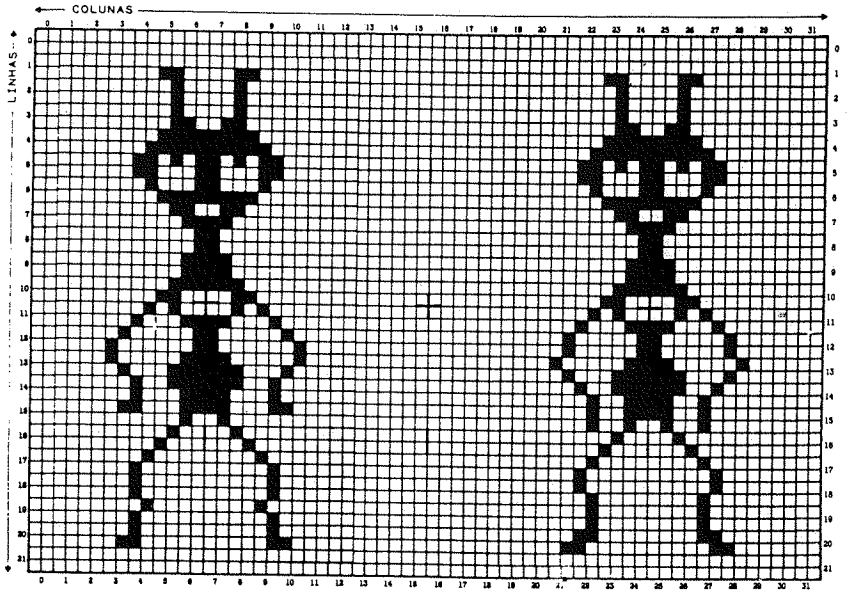

COLUNAS
LINHAS

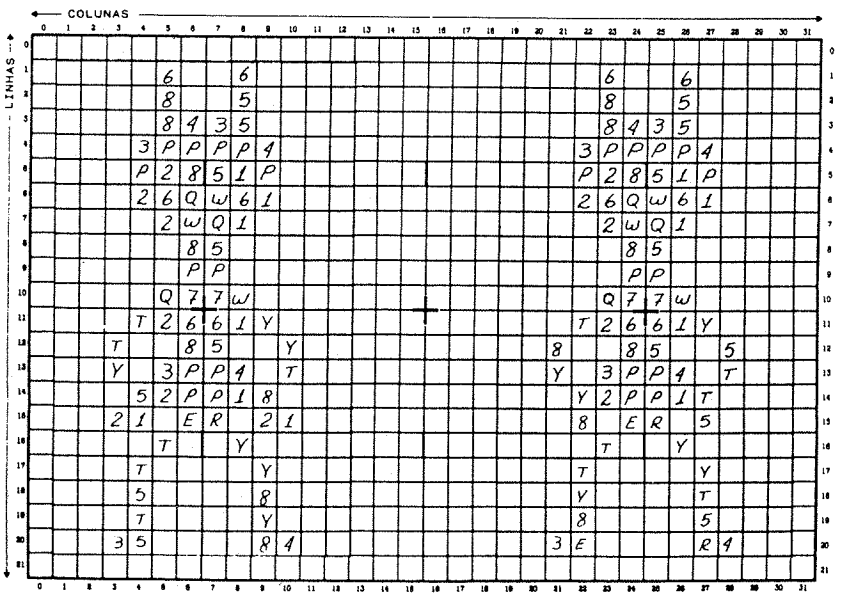

COLUNAS
LINHAS

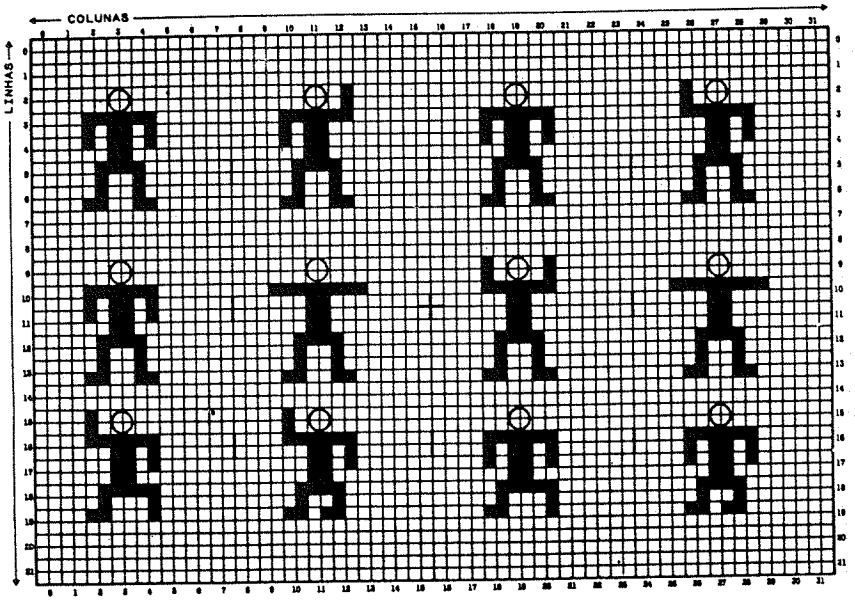

COLUNAS
LINHAS

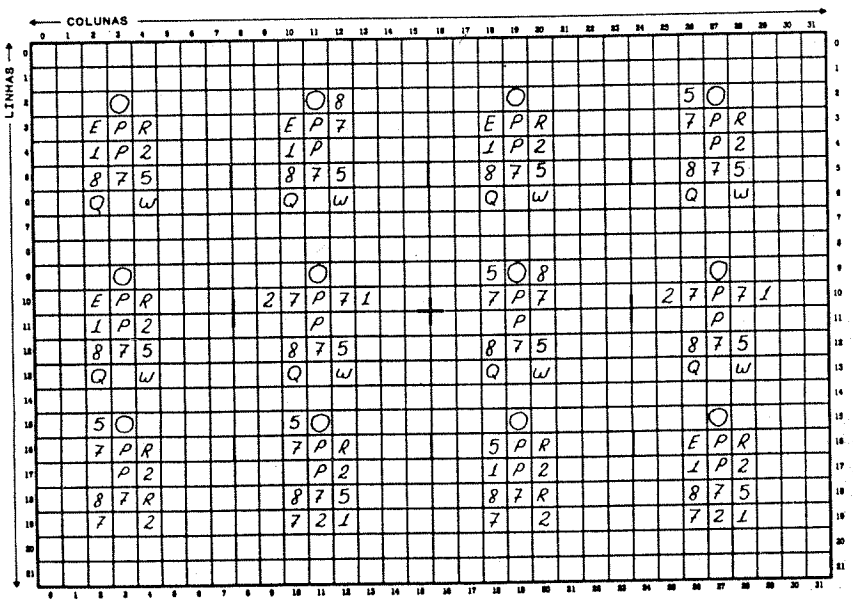

COLUNAS
LINHAS

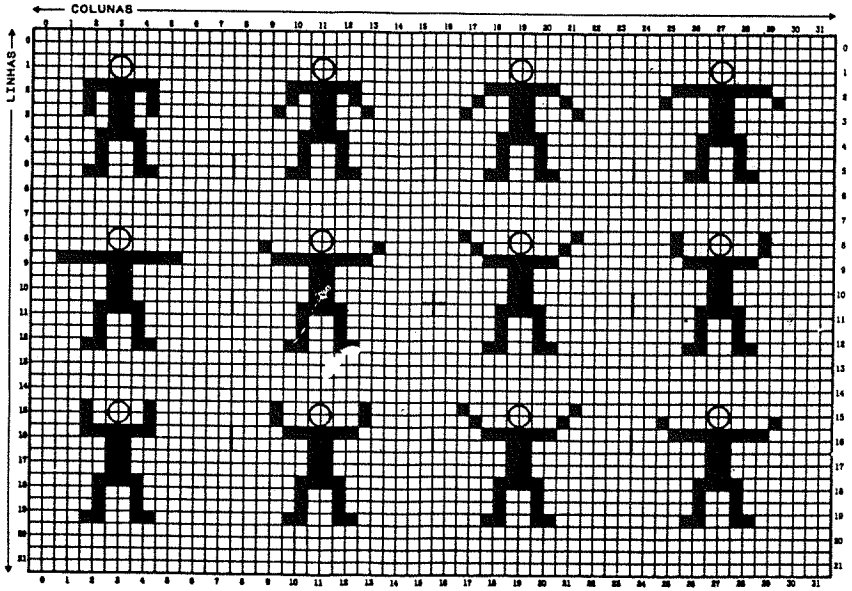

COLUNAS
LINHAS

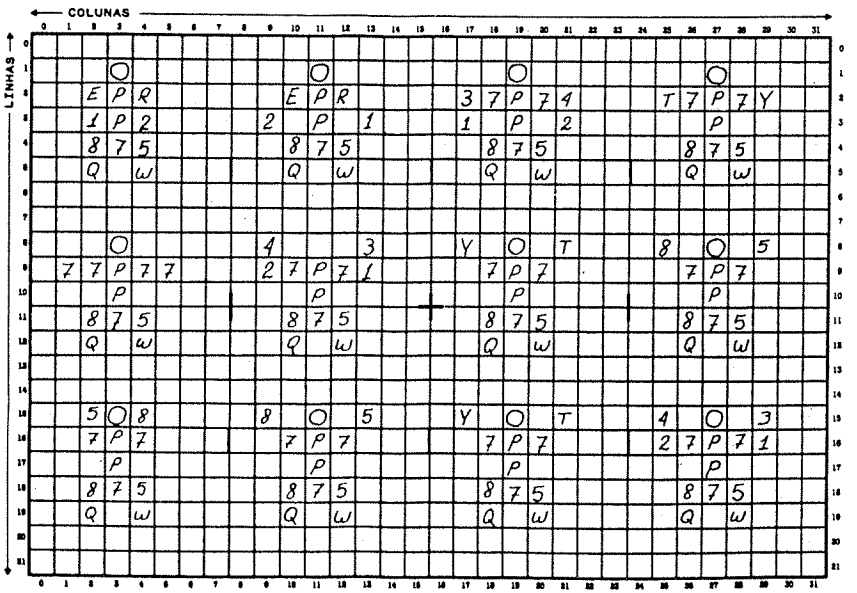

COLUNAS
LINHAS

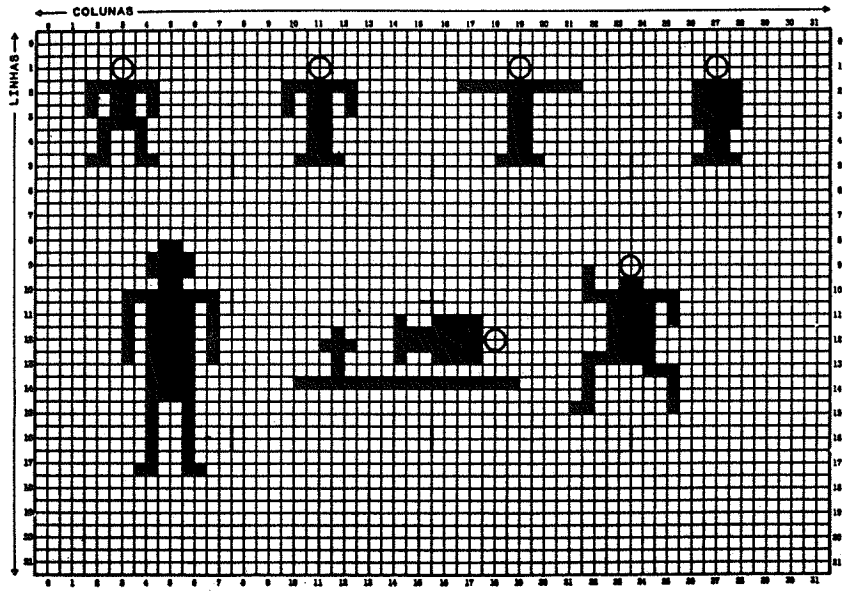

COLUNAS
LINHAS

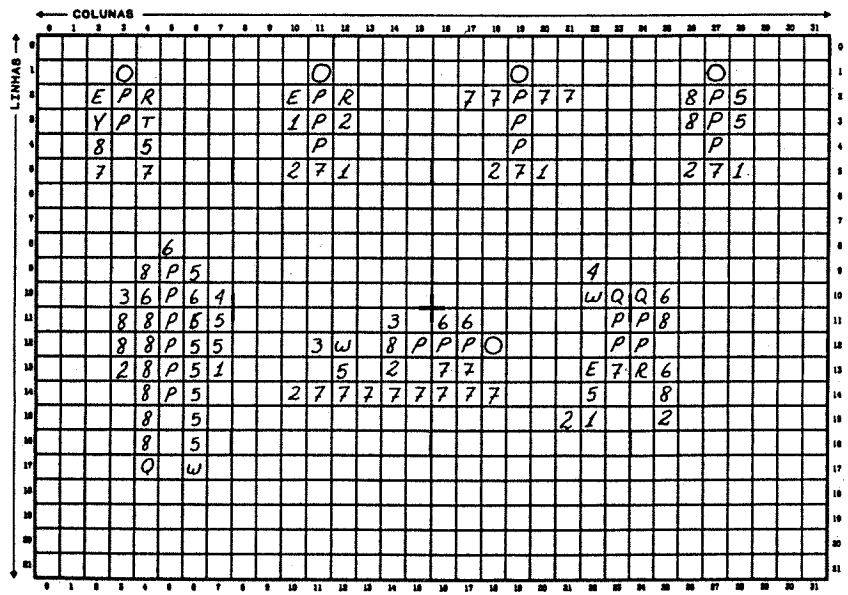

COLUNAS
LINHAS

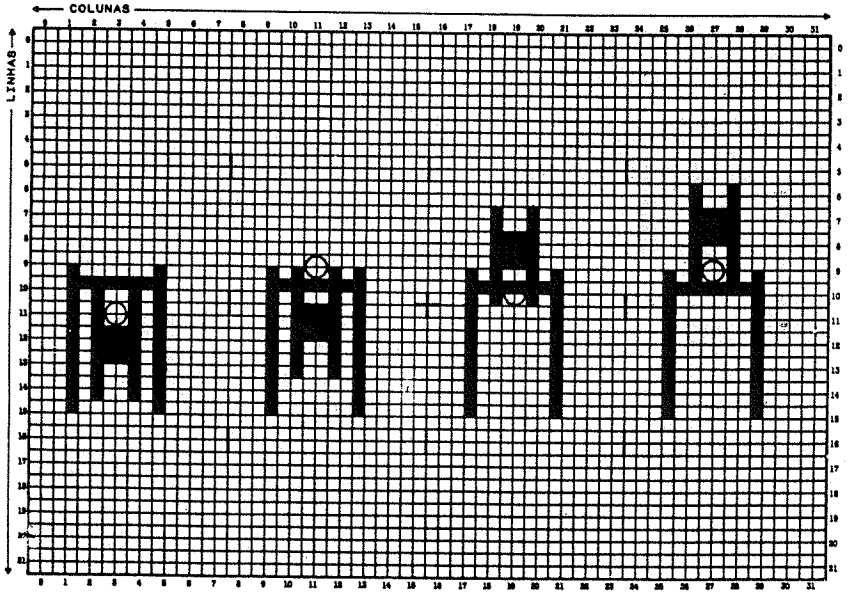

COLUNAS
LINHAS

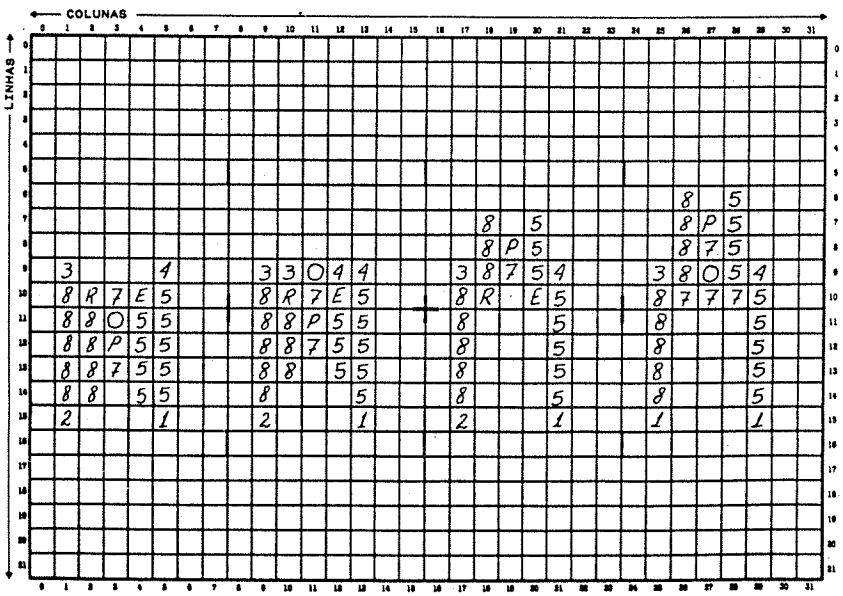

COLUNAS
LINHAS

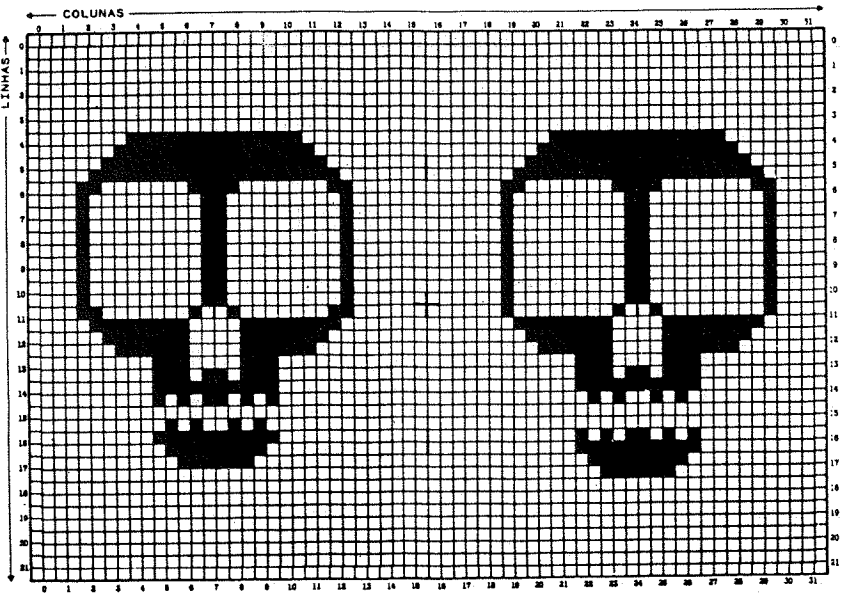

COLUNAS
LINHAS

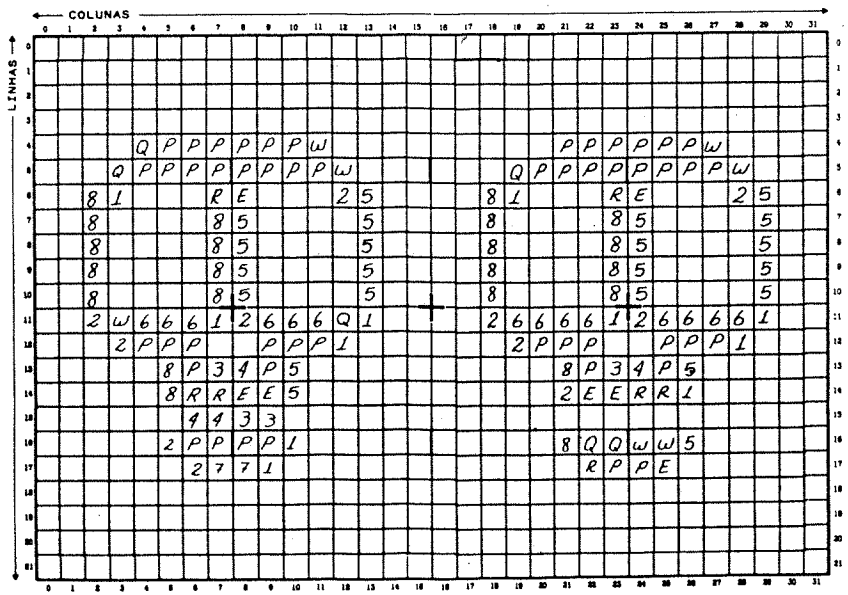

COLUNAS
LINHAS

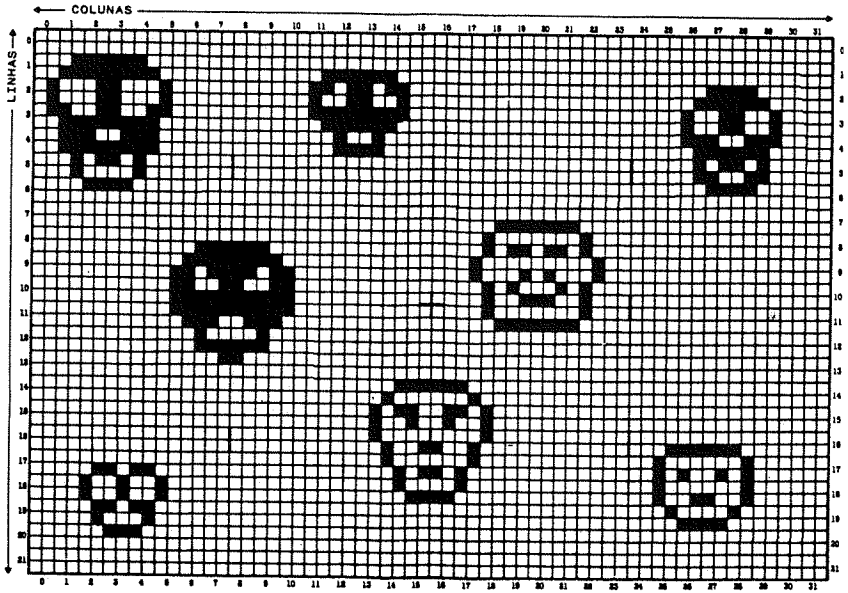

COLUNAS
LINHAS

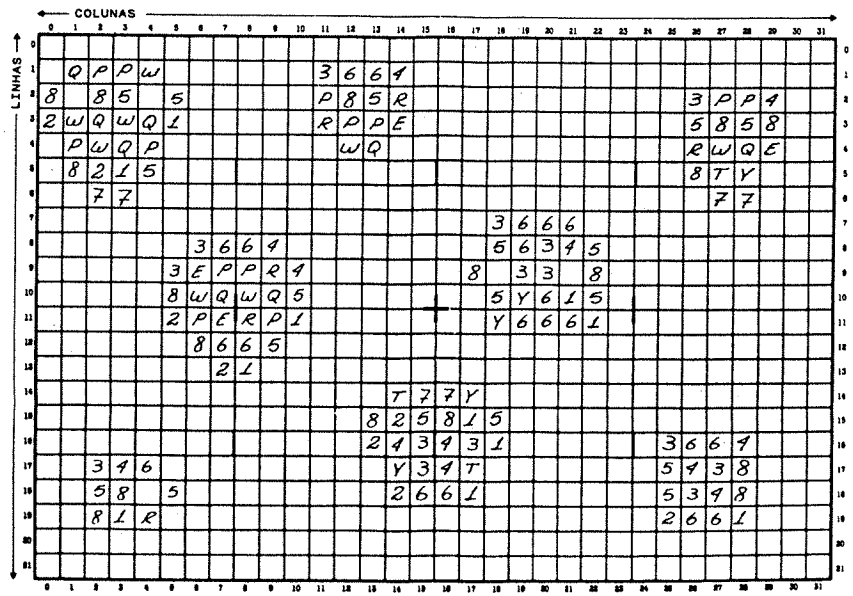

COLUNAS
LINHAS

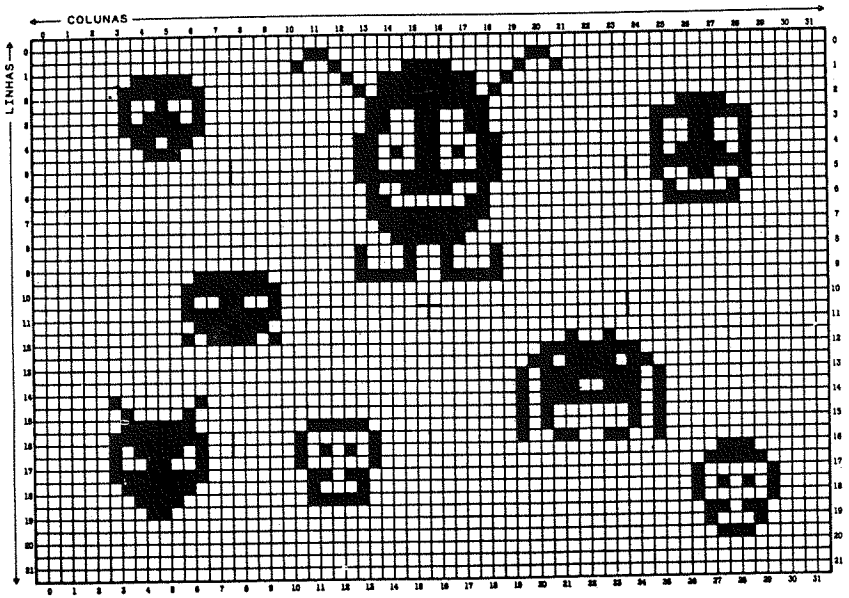

COLUNAS
LINHAS

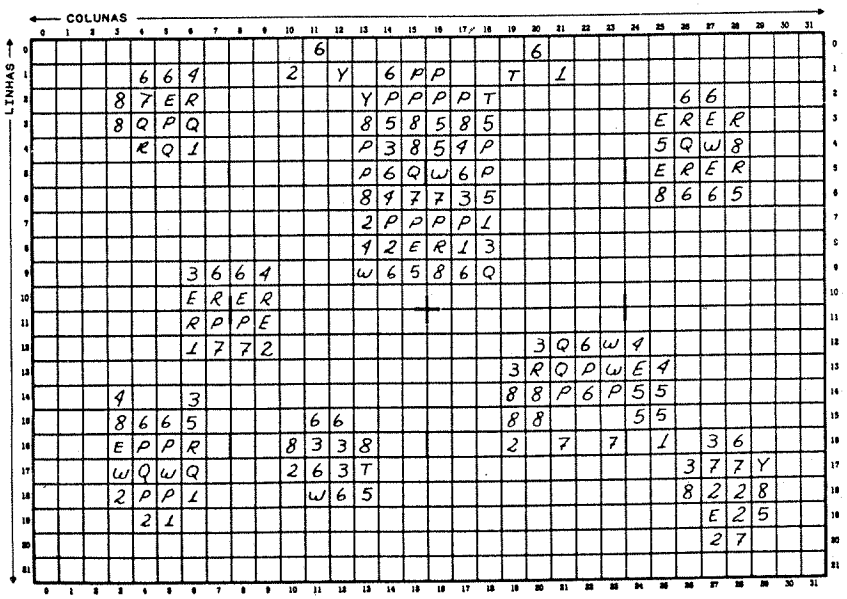

COLUNAS
LINHAS

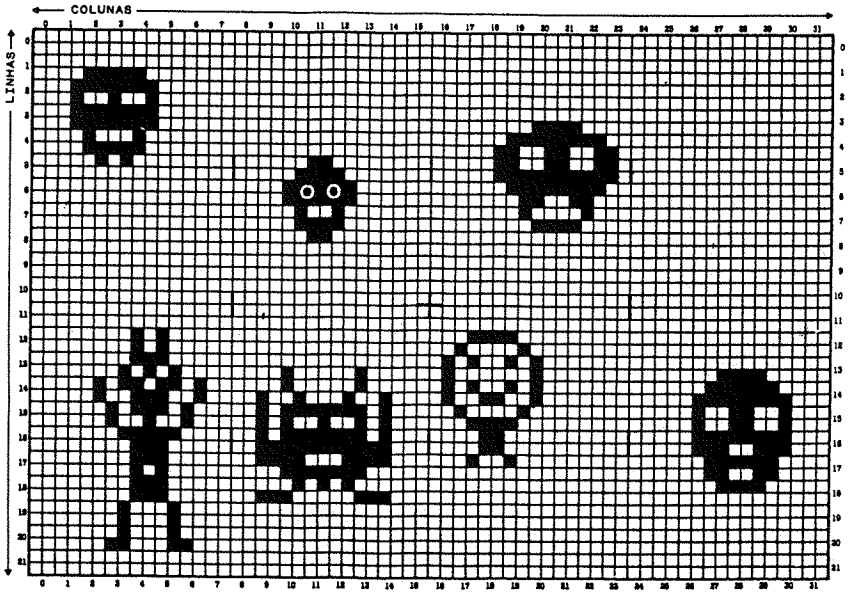

COLUNAS
LINHAS

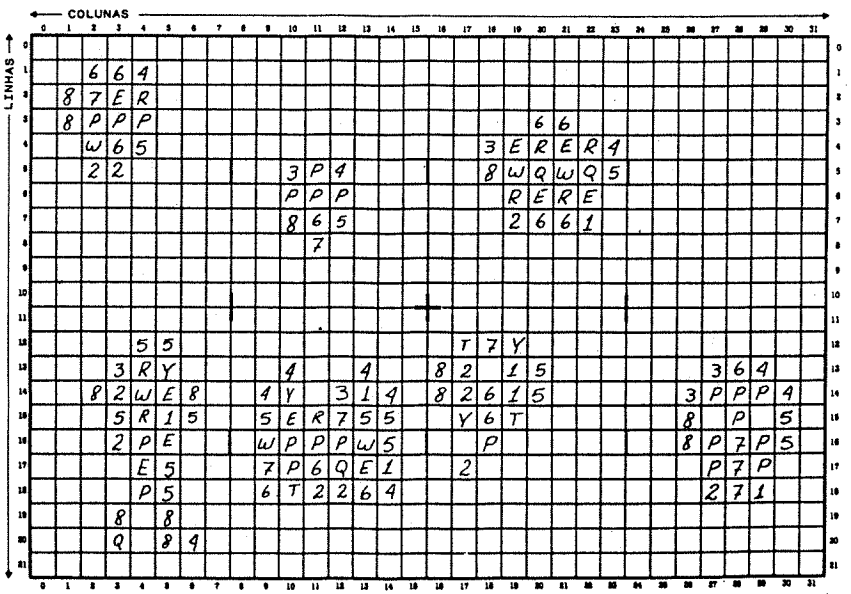

COLUNAS
LINHAS

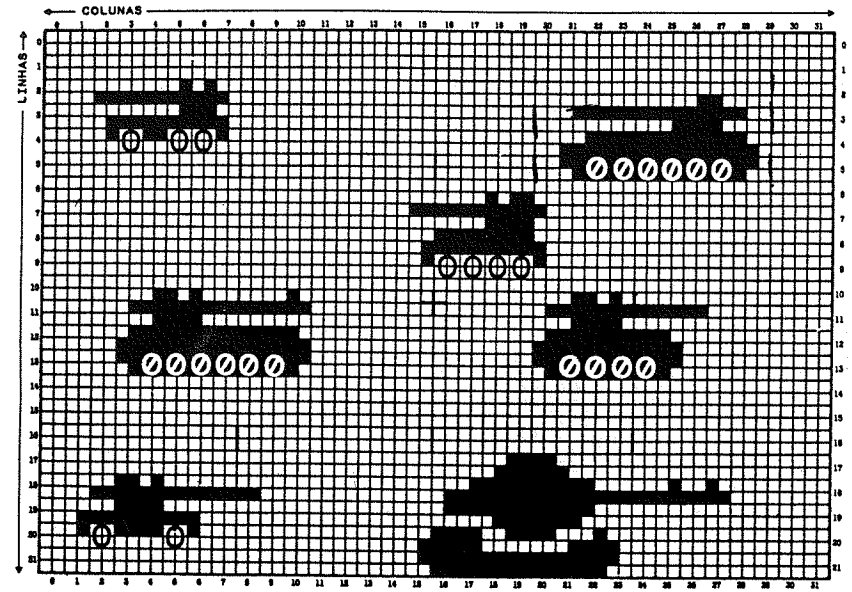

COLUNAS
LINHAS

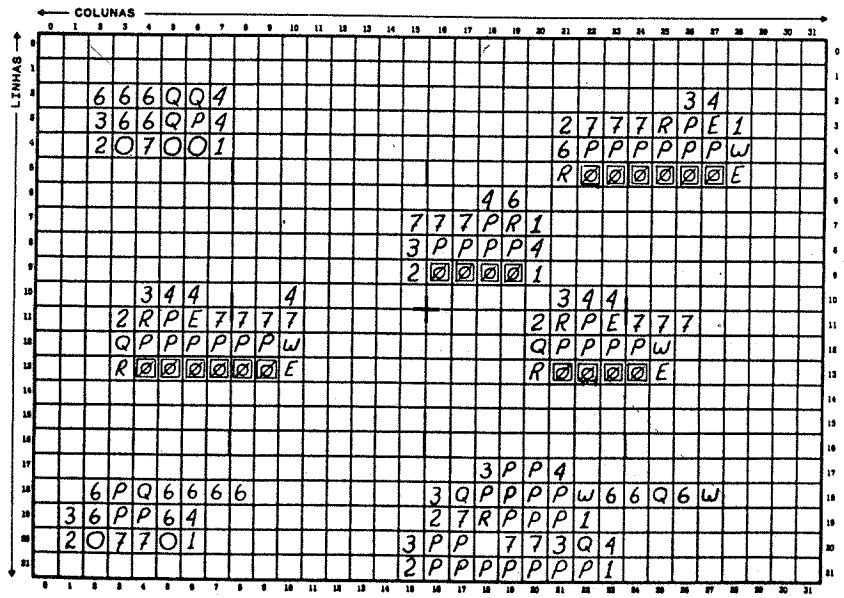

COLUNAS
LINHAS

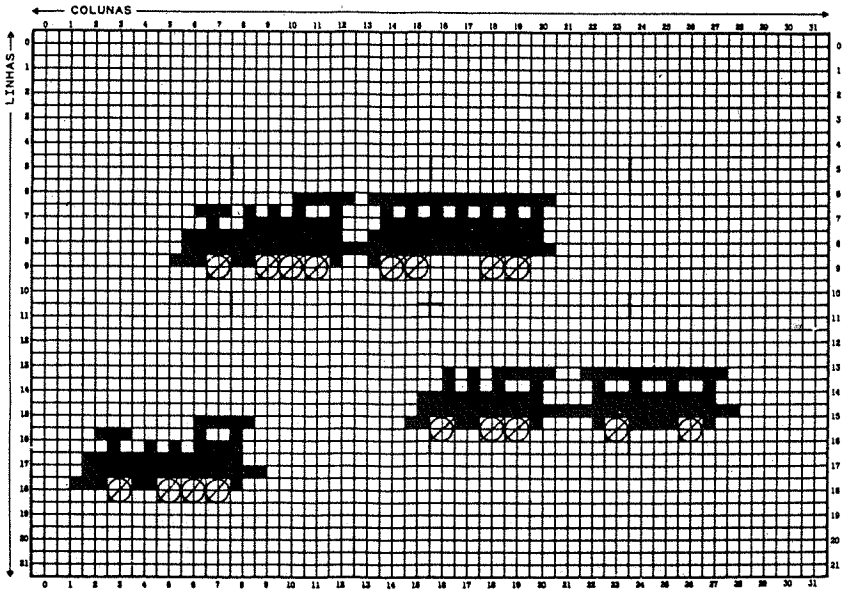

COLUNAS
LINHAS

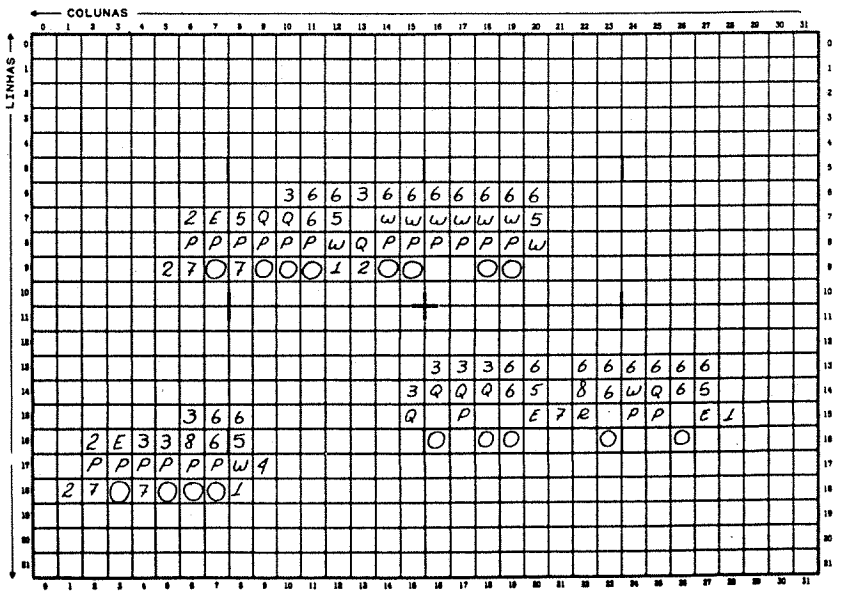

COLUNAS
LINHAS

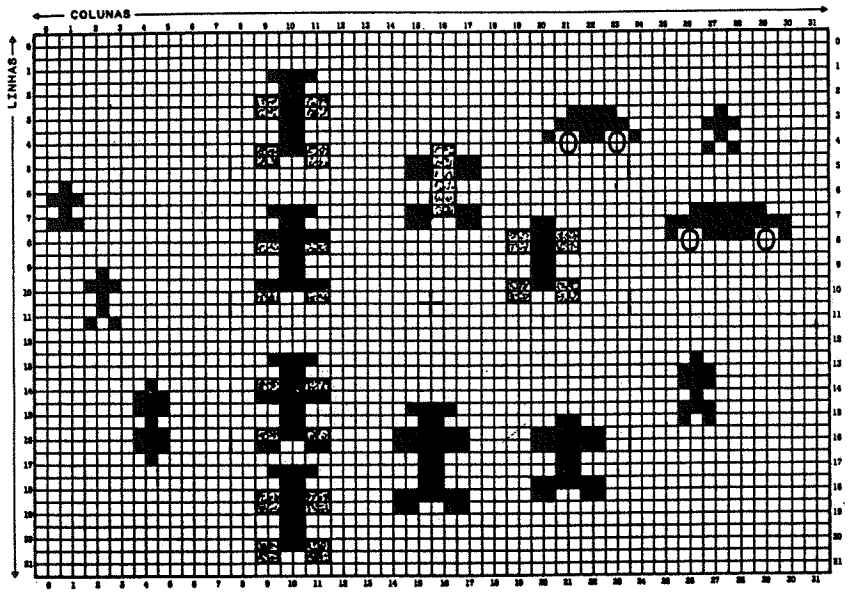

COLUNAS
LINHAS

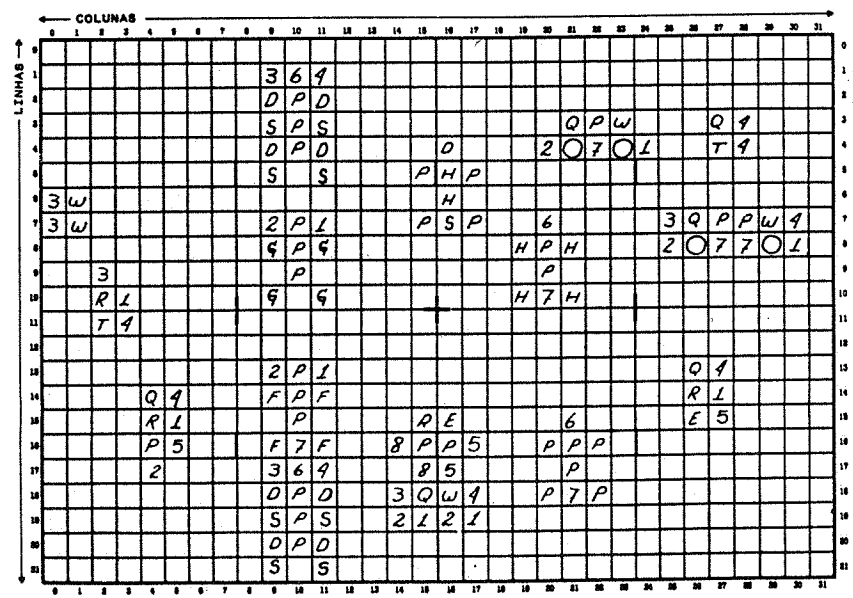

COLUNAS
LINHAS

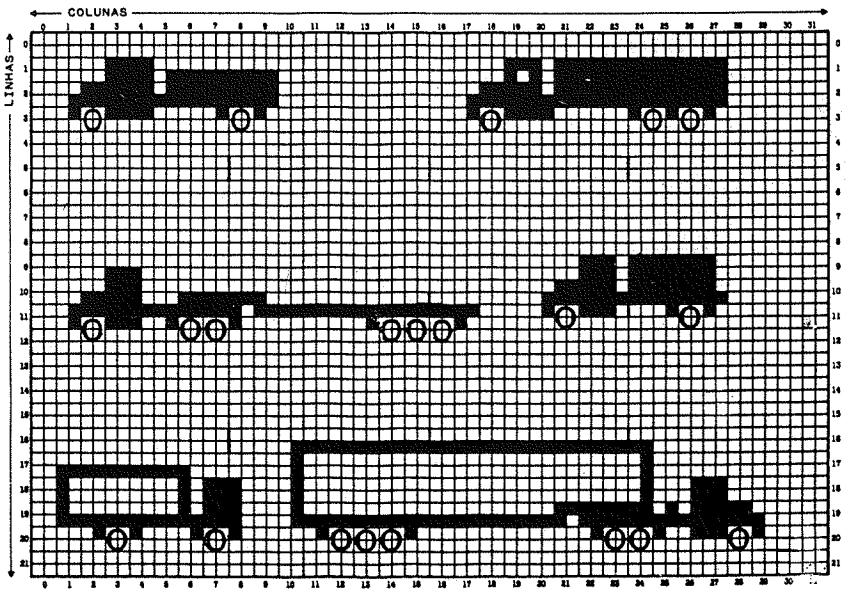

COLUNAS
LINHAS

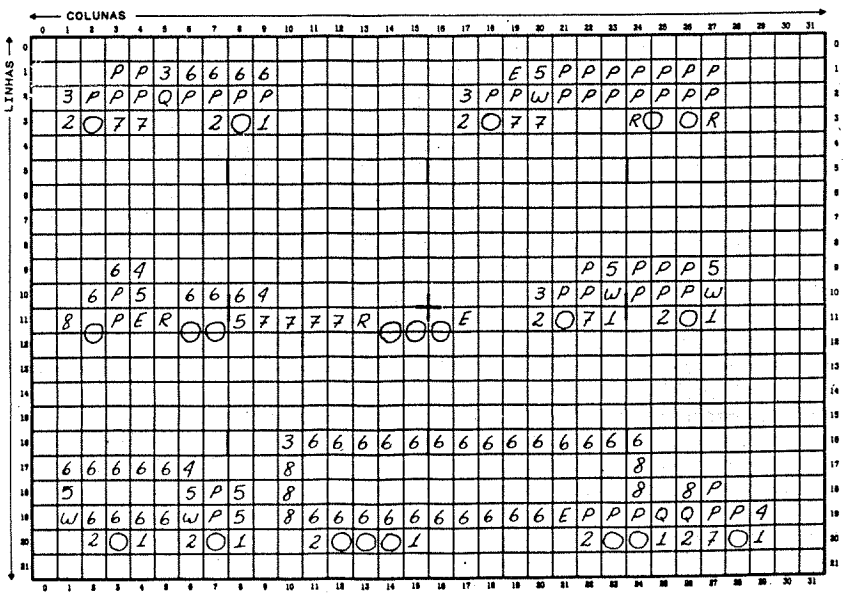

COLUNAS
LINHAS

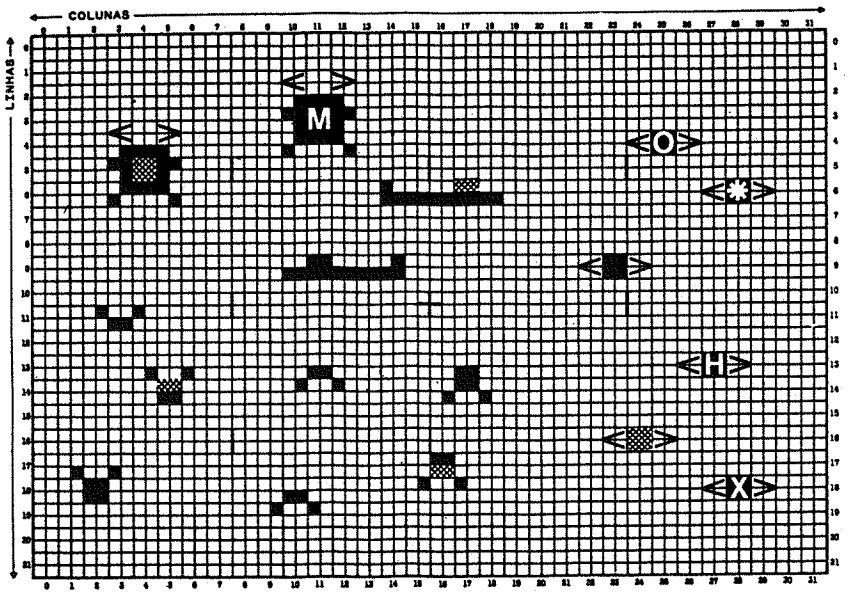

COLUNAS
LINHAS

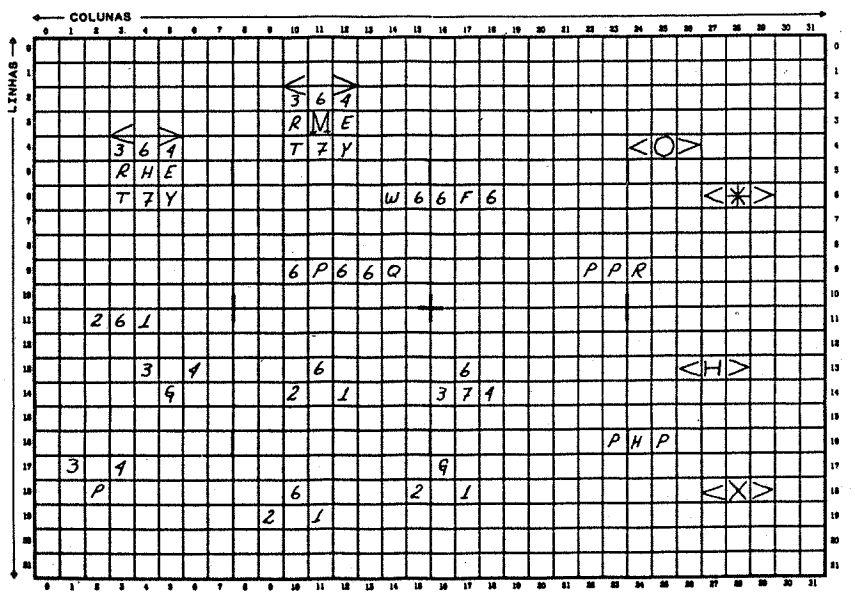

COLUNAS
LINHAS

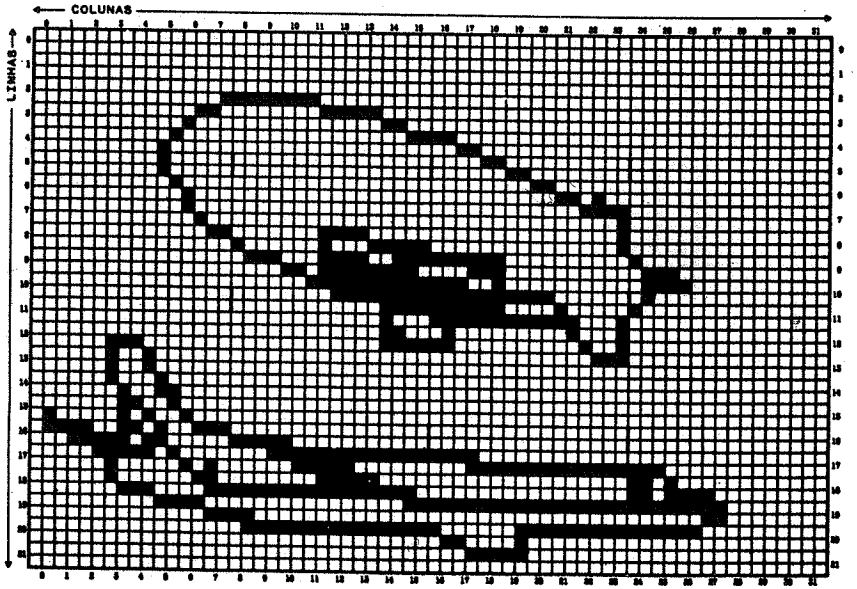

COLUNAS
LINHAS

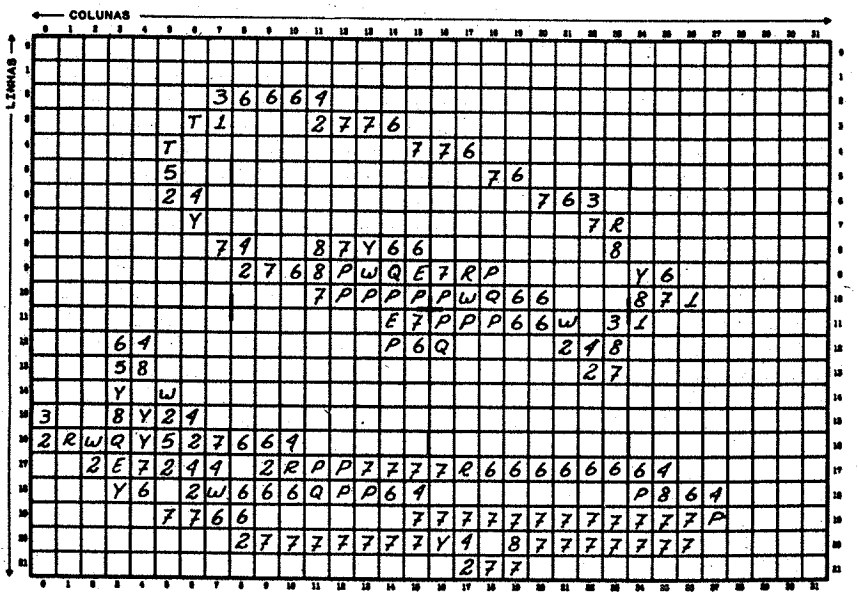

COLUNAS
LINHAS

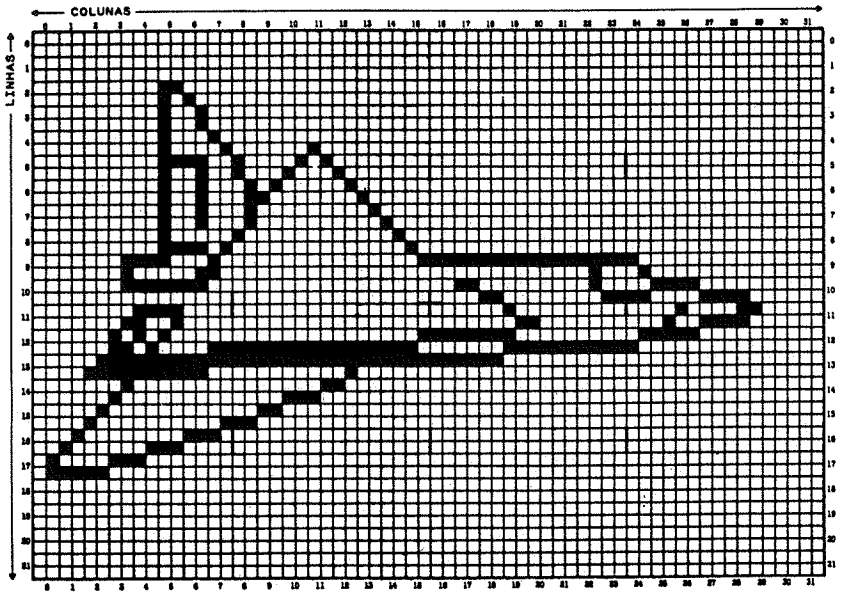

COLUNAS
LINHAS

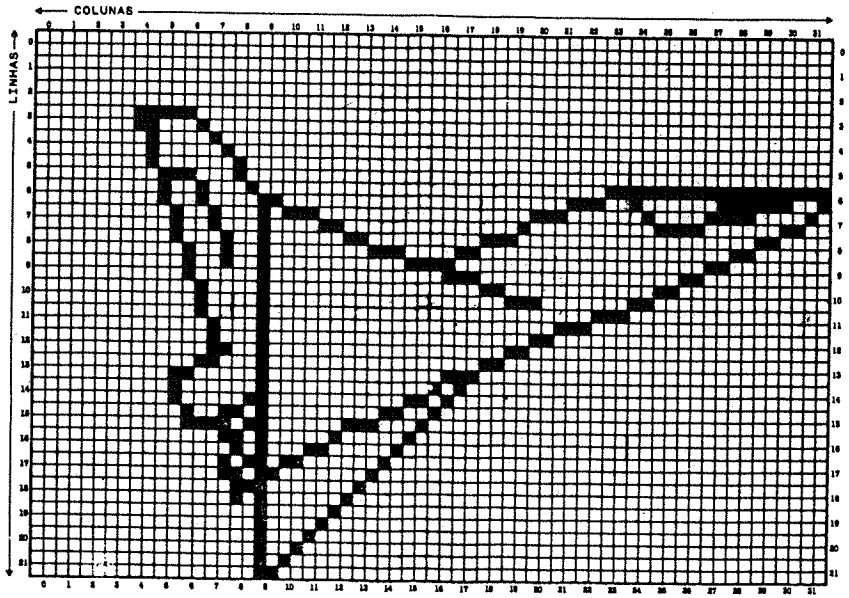

COLUNAS
LINHAS
M
O
*
H
X

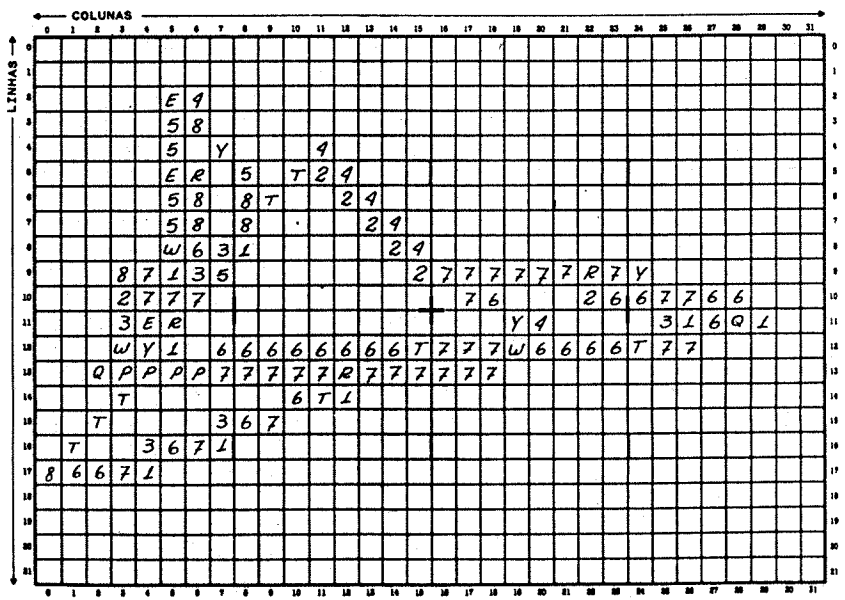

COLUNAS
LINHAS

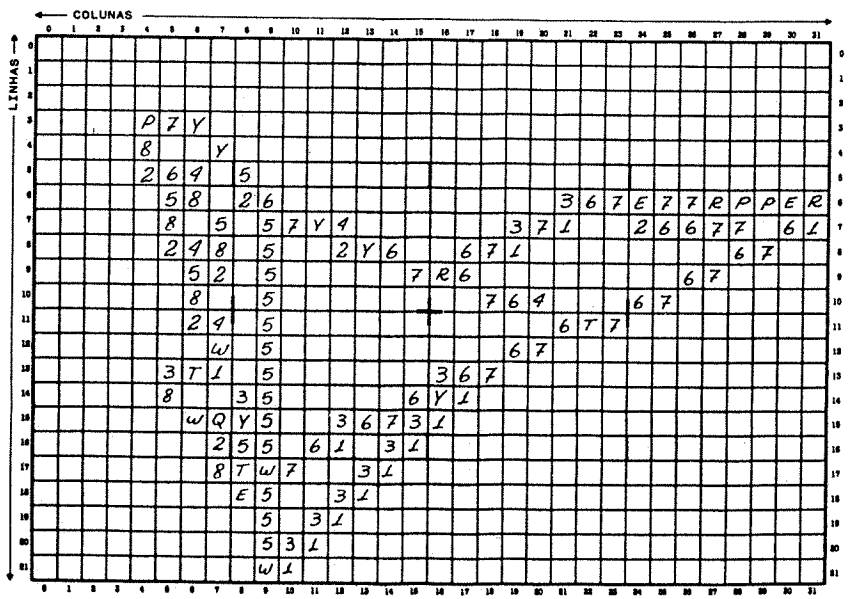

COLUNAS
LINHAS

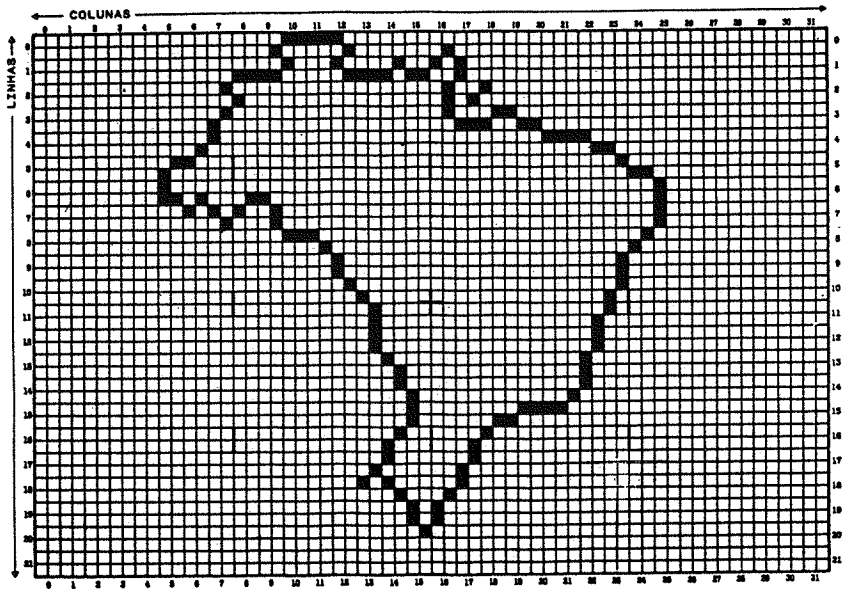

COLUNAS
LINHAS

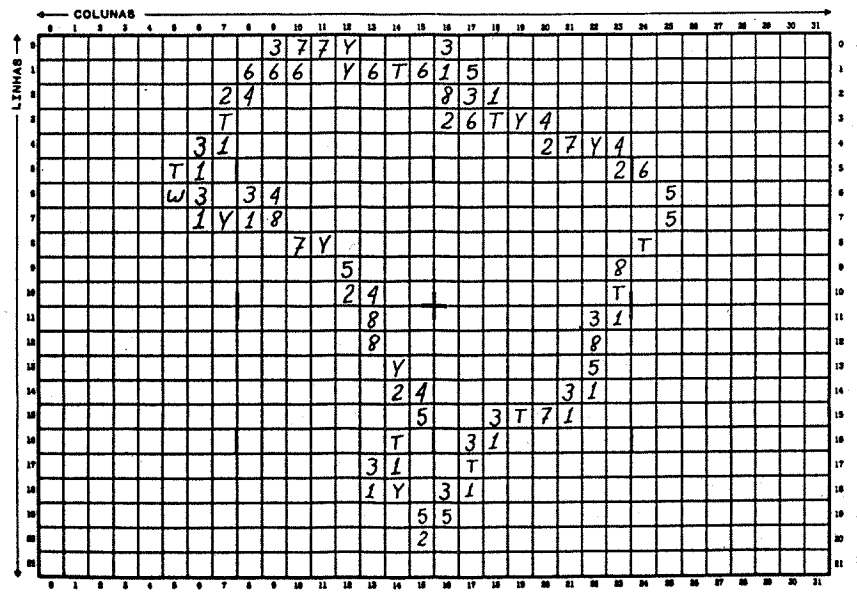

COLUNAS
LINHAS

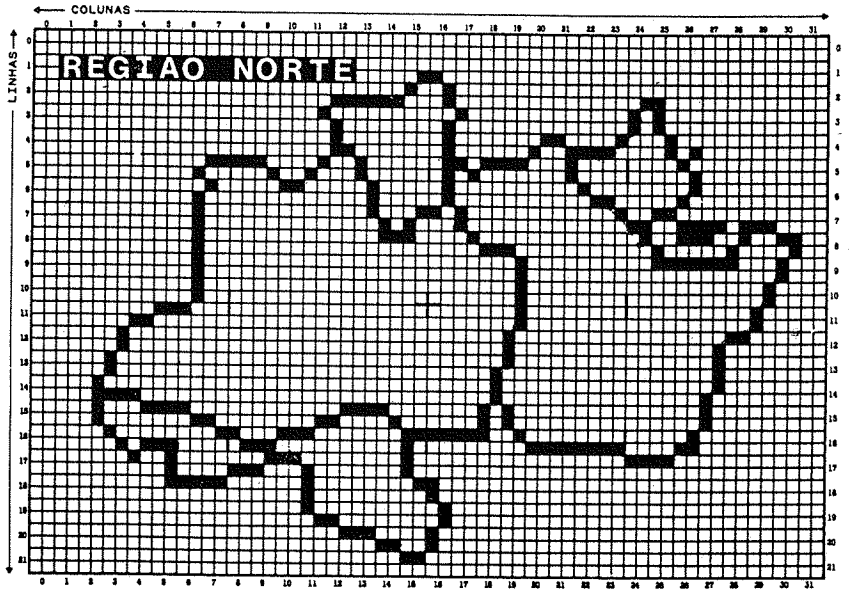

COLUNAS
LINHAS

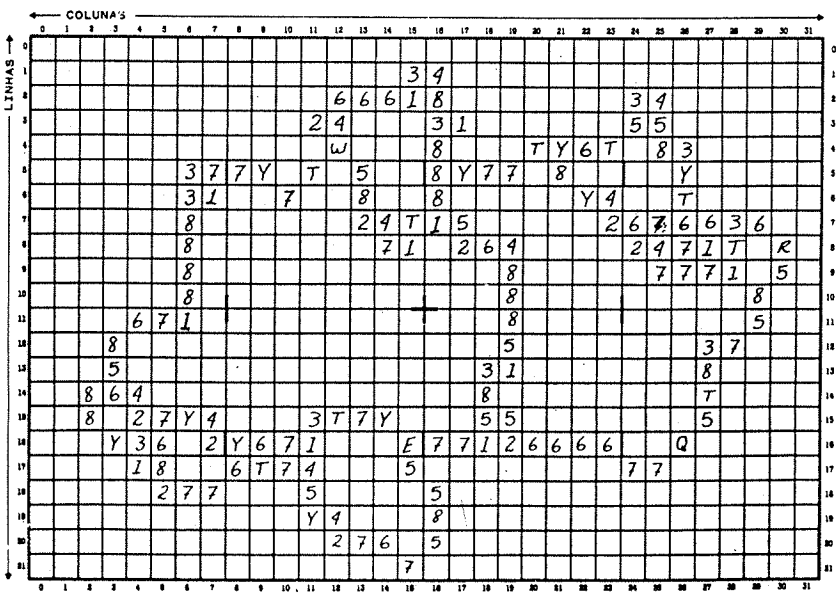

COLUNAS
LINHAS

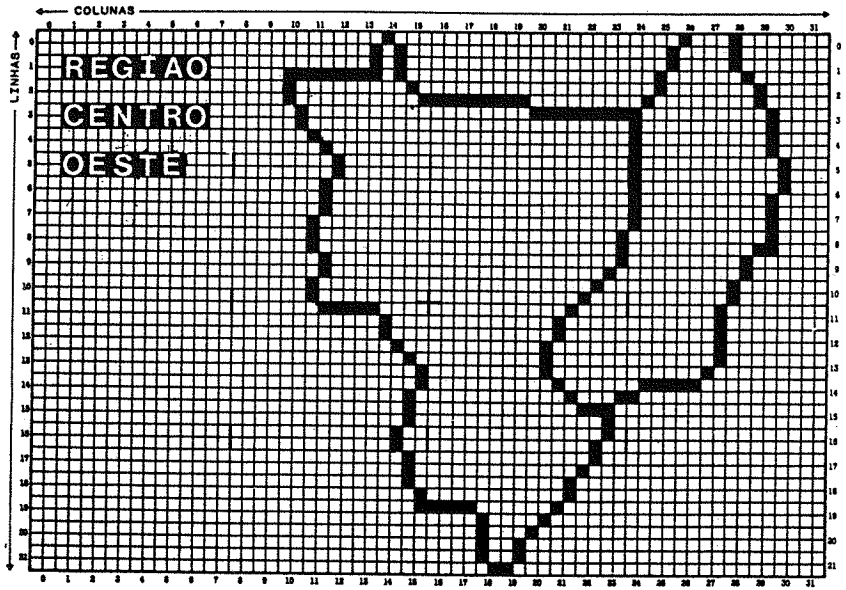

COLUNAS
LINHAS

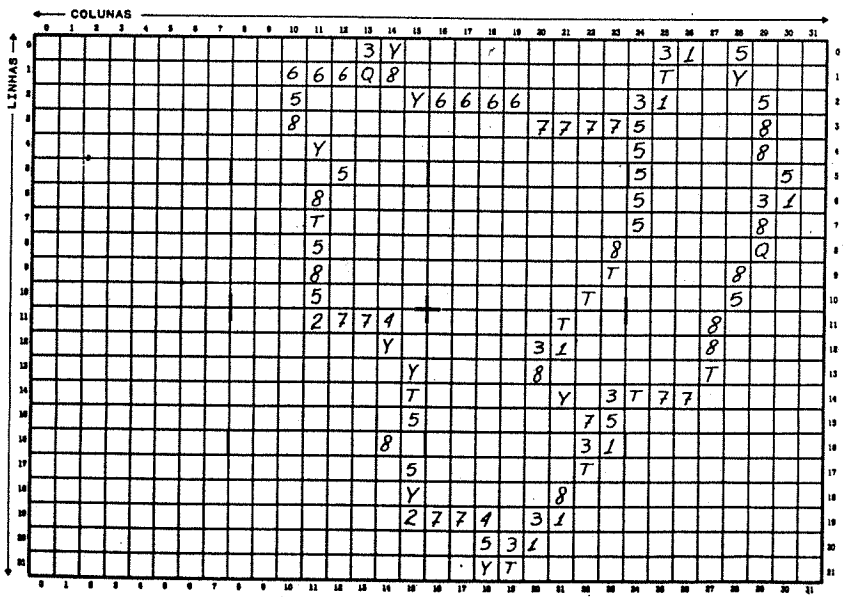

COLUNAS
LINHAS

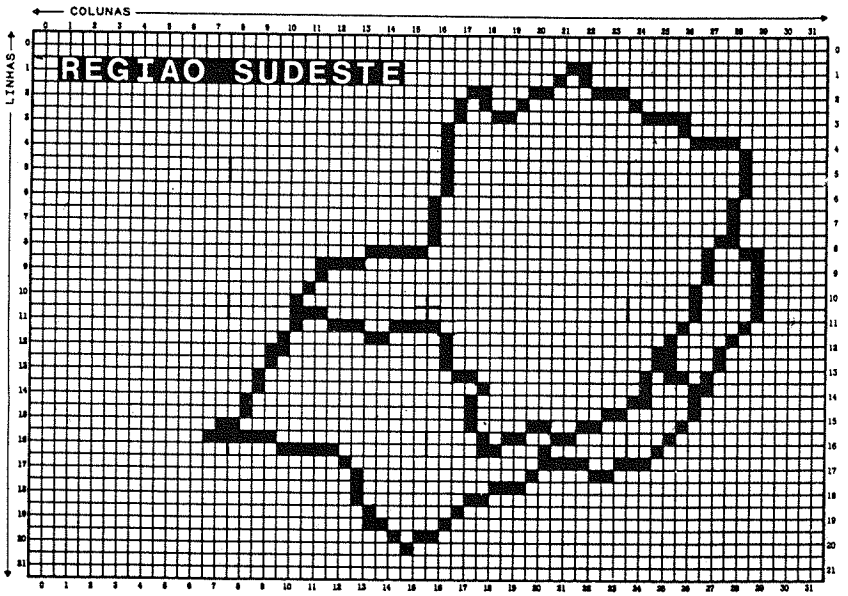

COLUNAS
LINHAS

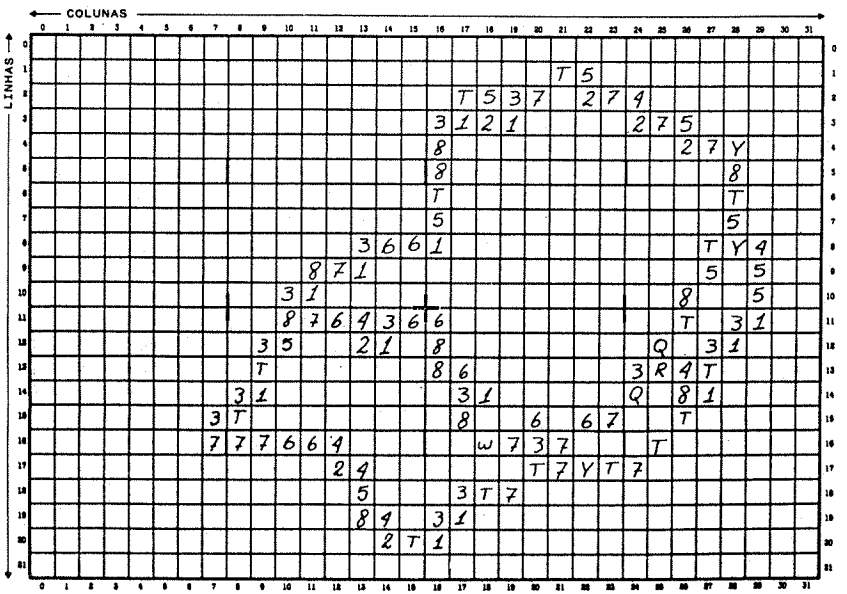

COLUNAS
LINHAS
REGIAO NORTE

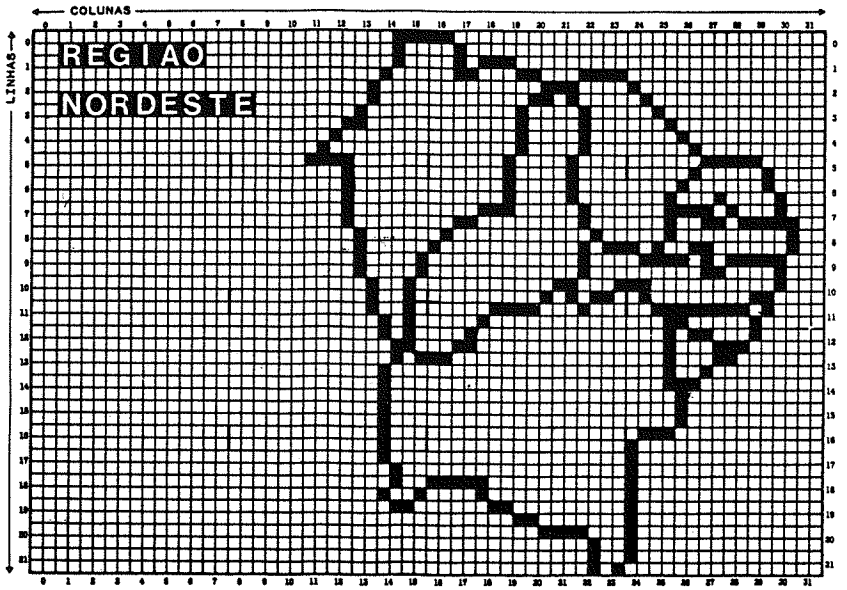

COLUNAS
LINHAS

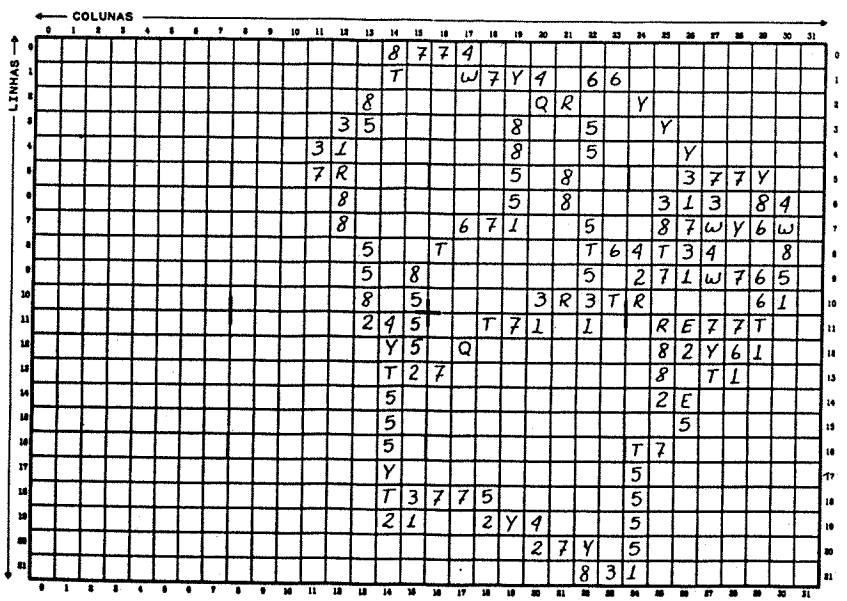

COLUNAS
LINHAS
REGIAO
CENTRO
OESTE

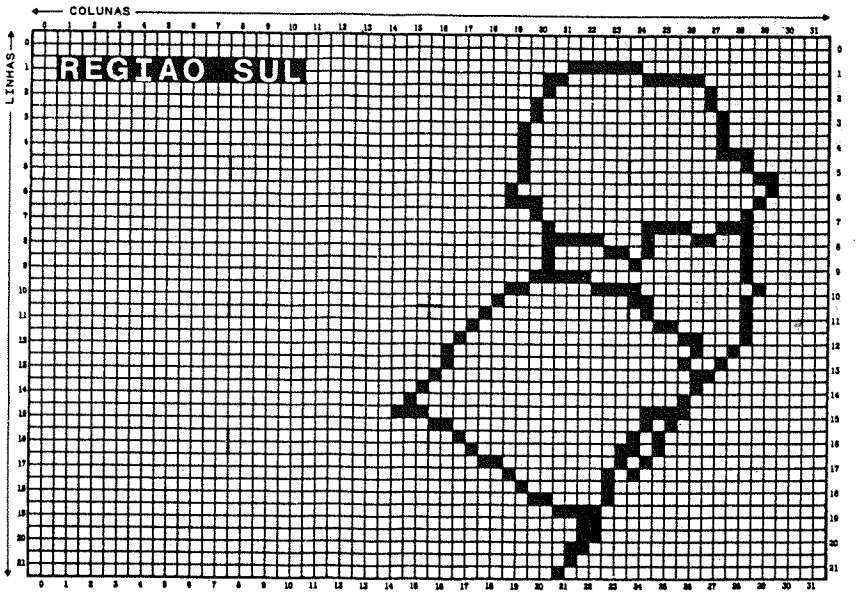

COLUNAS
LINHAS

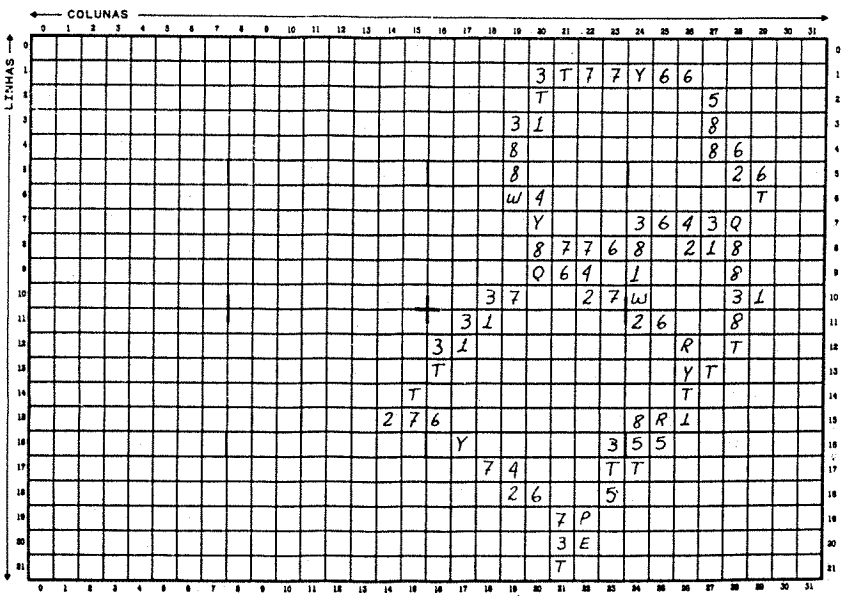

COLUNAS
LINHAS
REGIAO SUDESTE

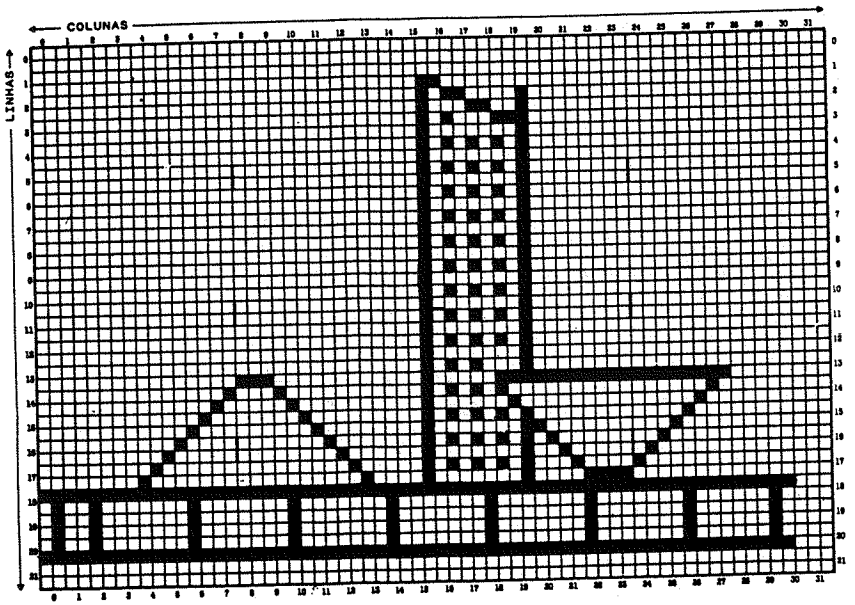

COLUNAS
LINHAS

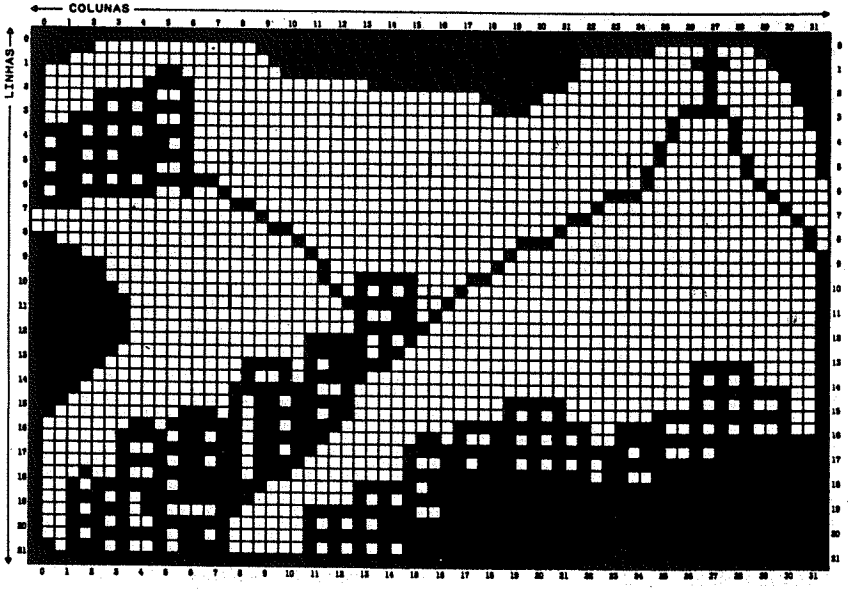

COLUNAS
LINHAS
REGIAO
NORDESTE

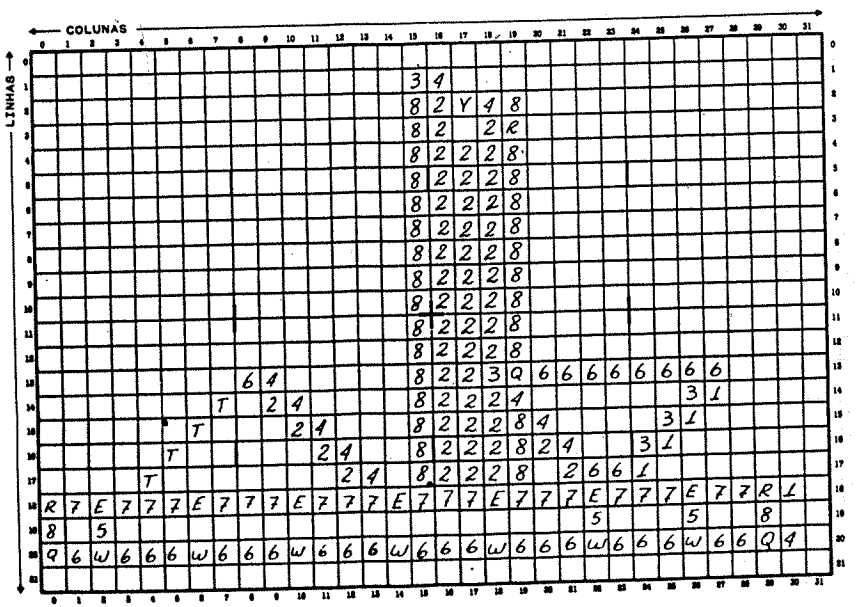

COLUNAS
LINHAS


COLUNAS
LINHAS
REGIAO SUL

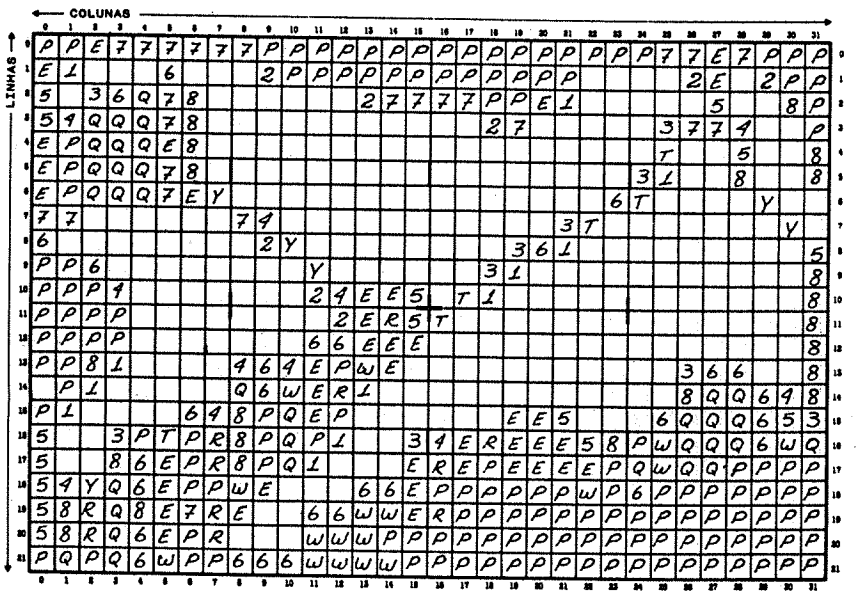

COLUNAS
LINHAS


COLUNAS
LINHAS

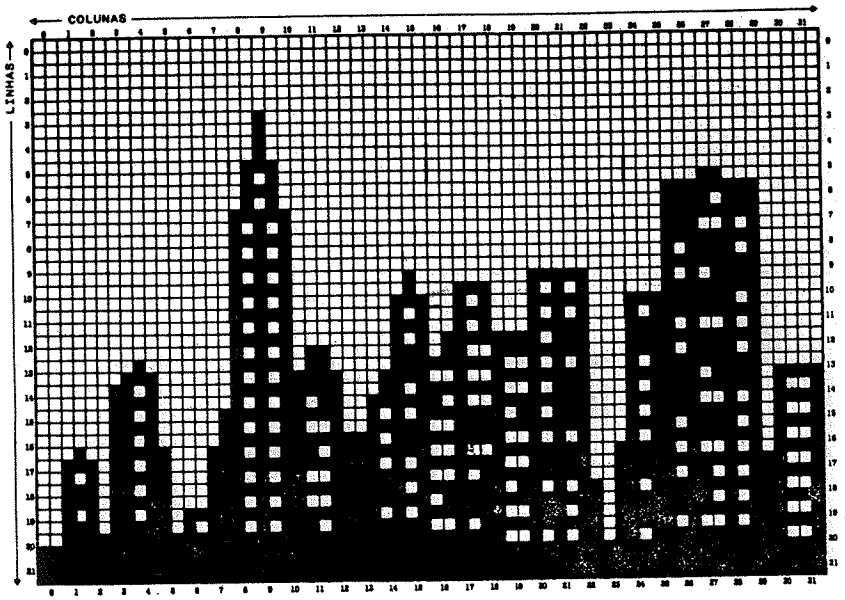

COLUNAS
LINHAS

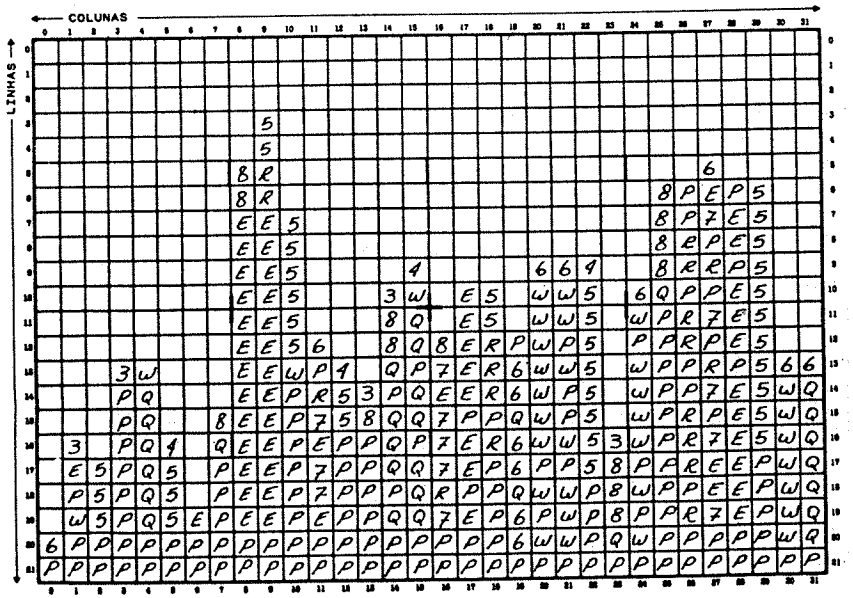

COLUNAS
LINHAS

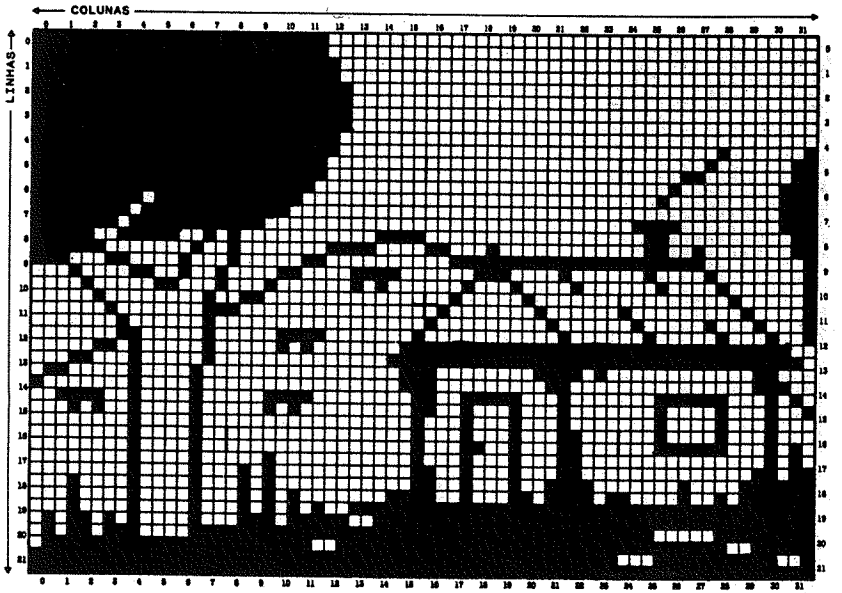
112

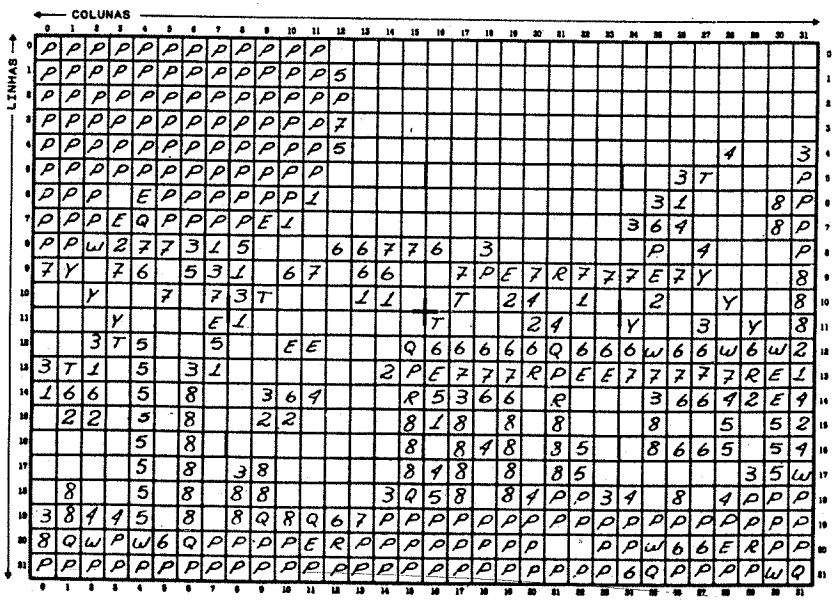

COLUNAS
LINHAS

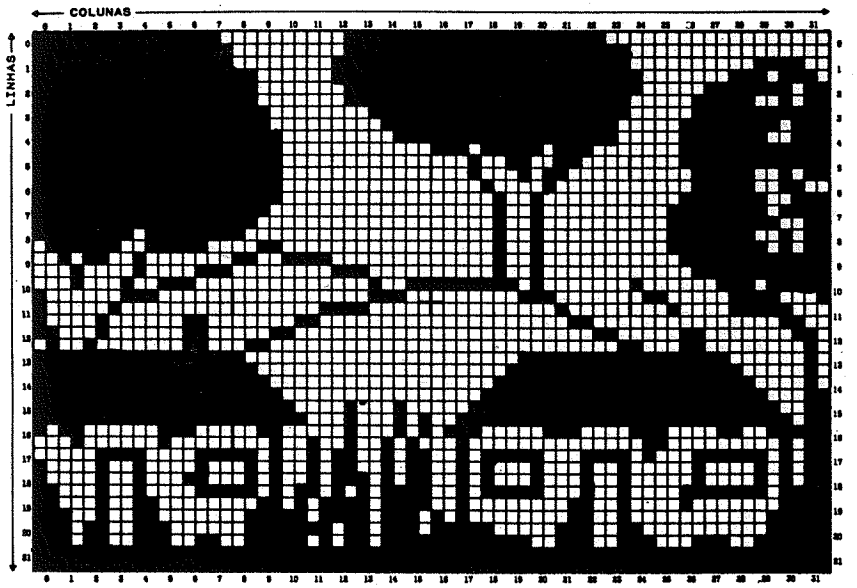
113

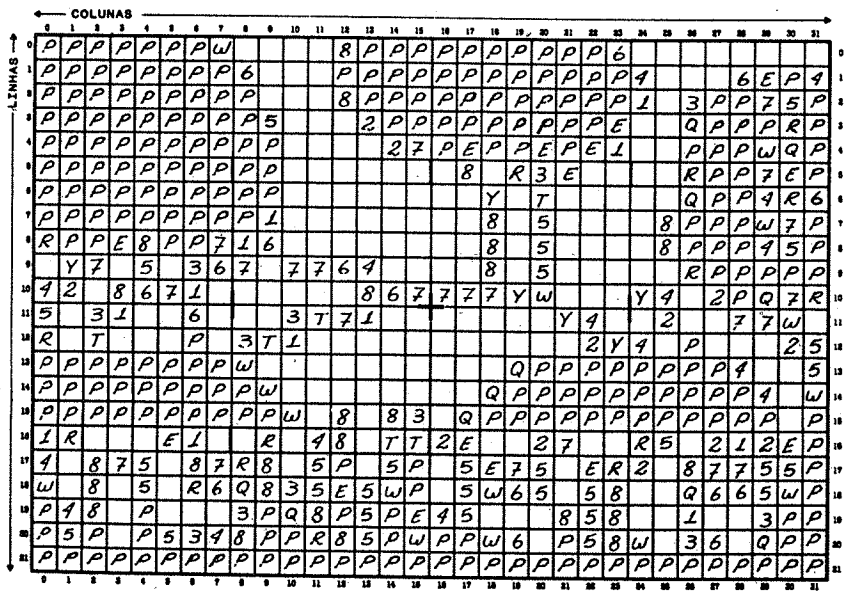

COLUNAS
LINHAS

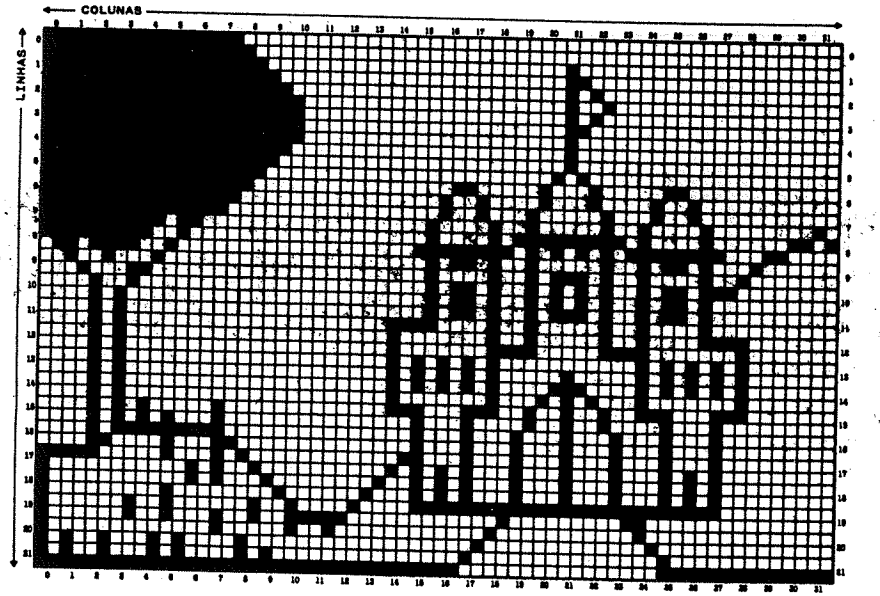

COLUNAS
LINHAS

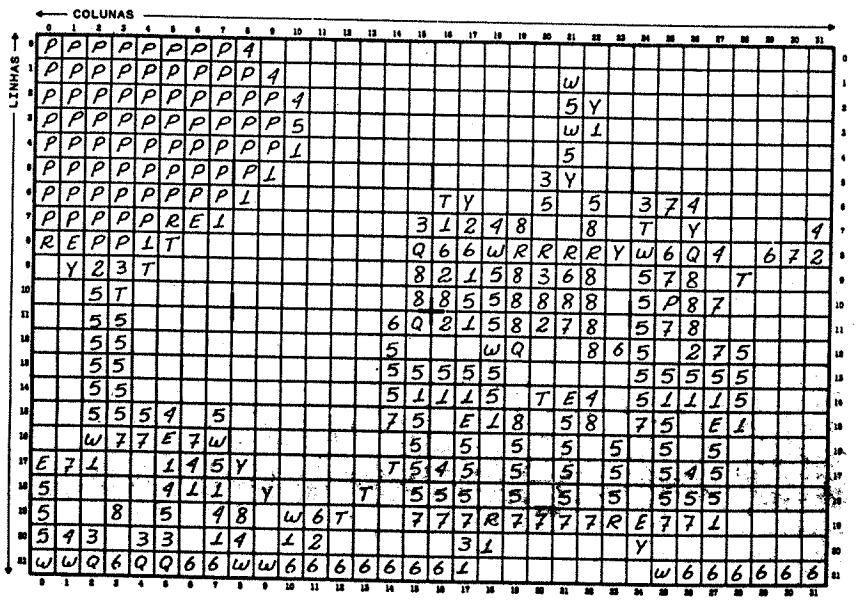

COLUNAS
LINHAS

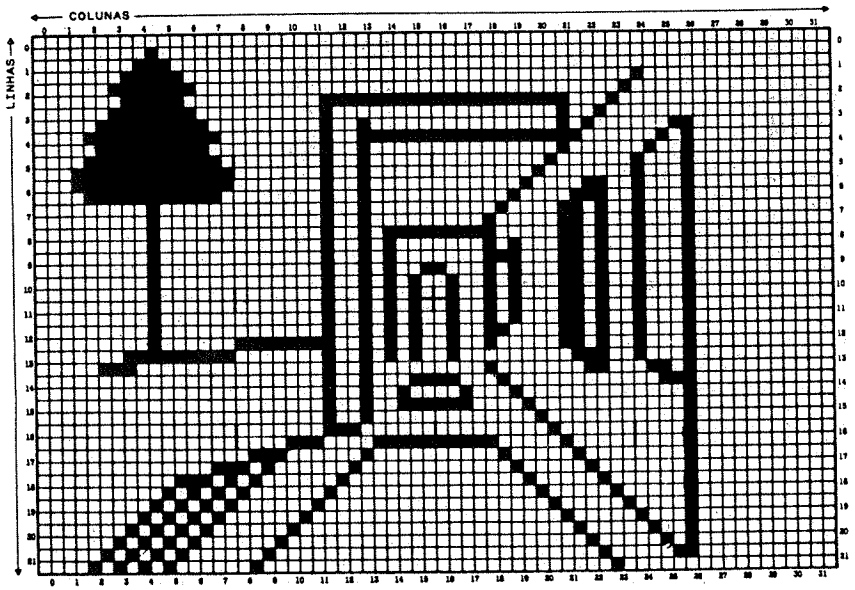

COLUNAS
LINHAS

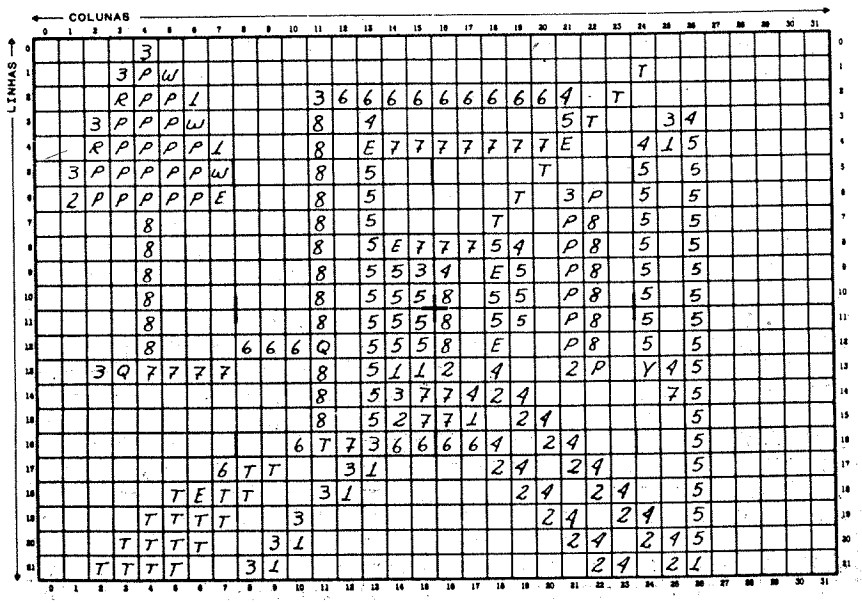

COLUNAS
LINHAS

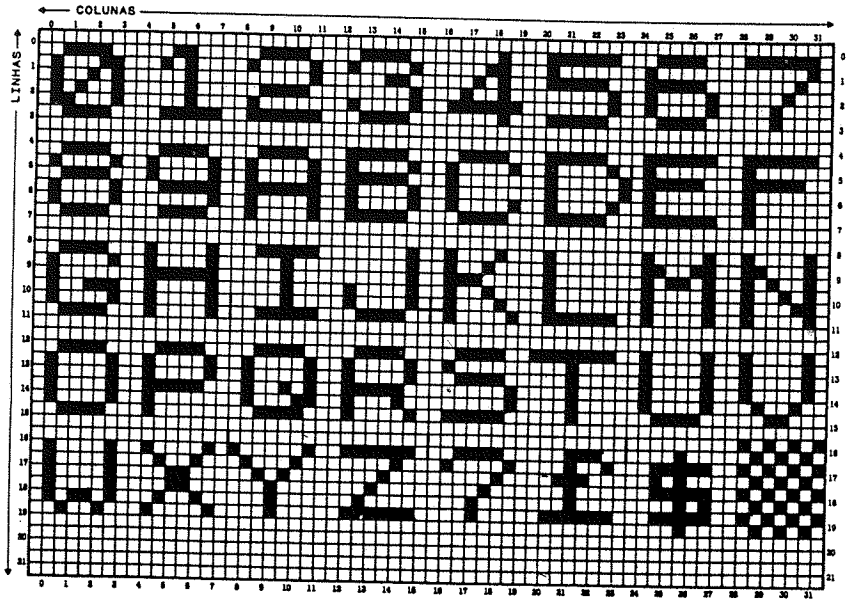

COLUNAS
LINHAS

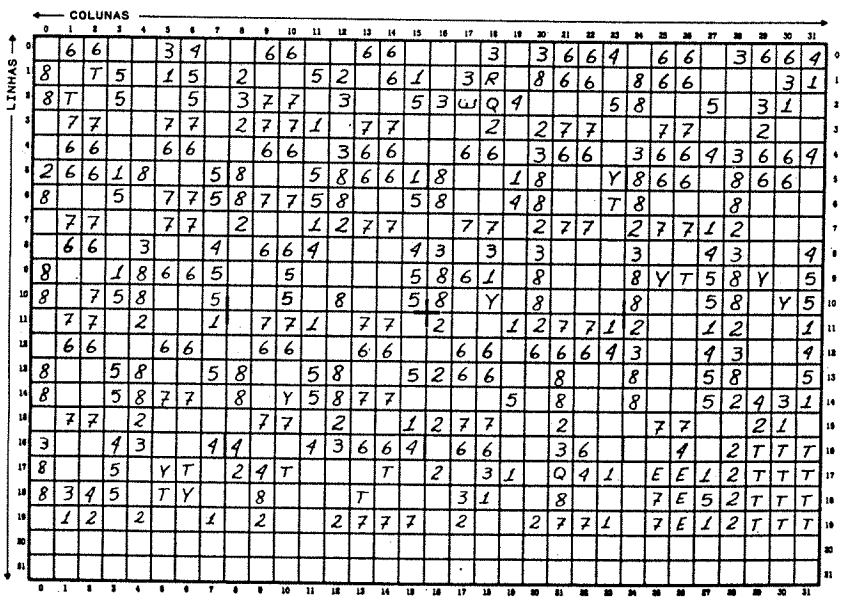

COLUNAS
LINHAS

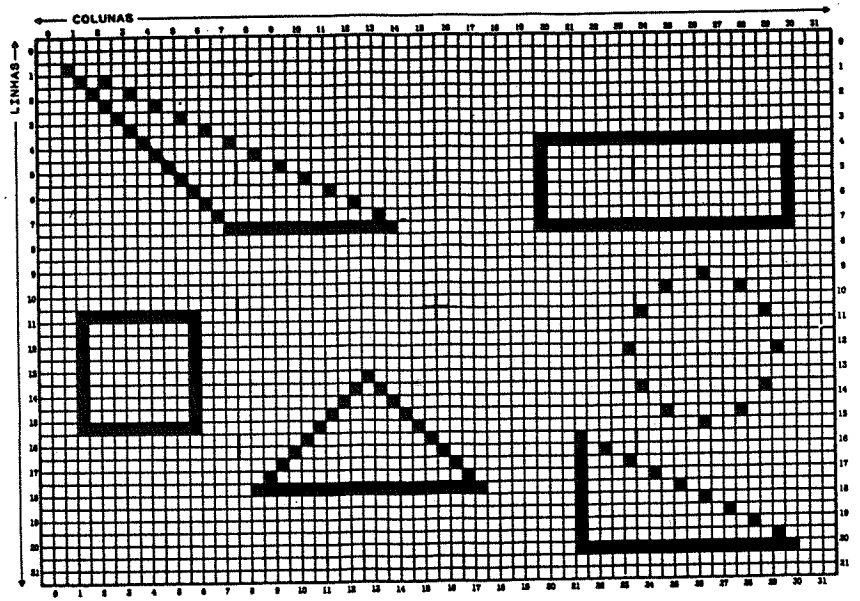

COLUNAS
LINHAS

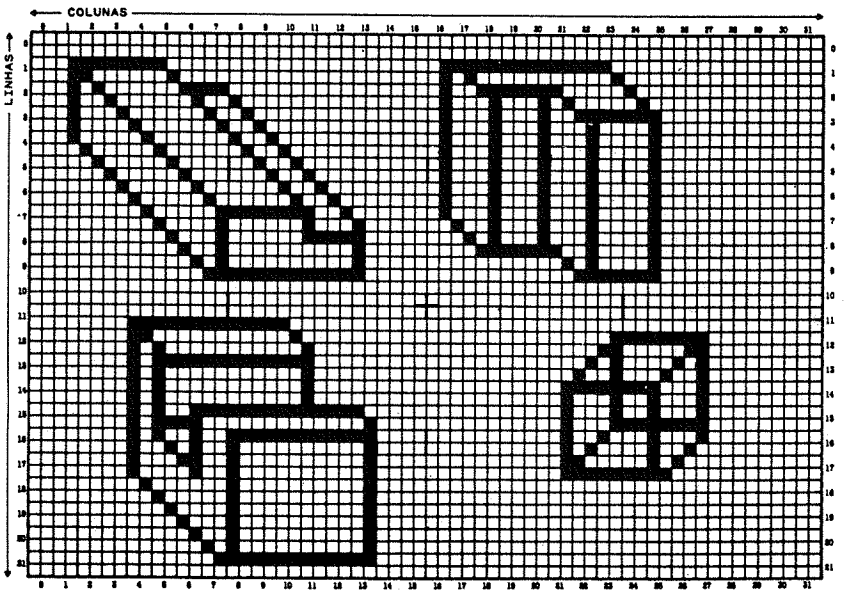

COLUNAS
LINHAS

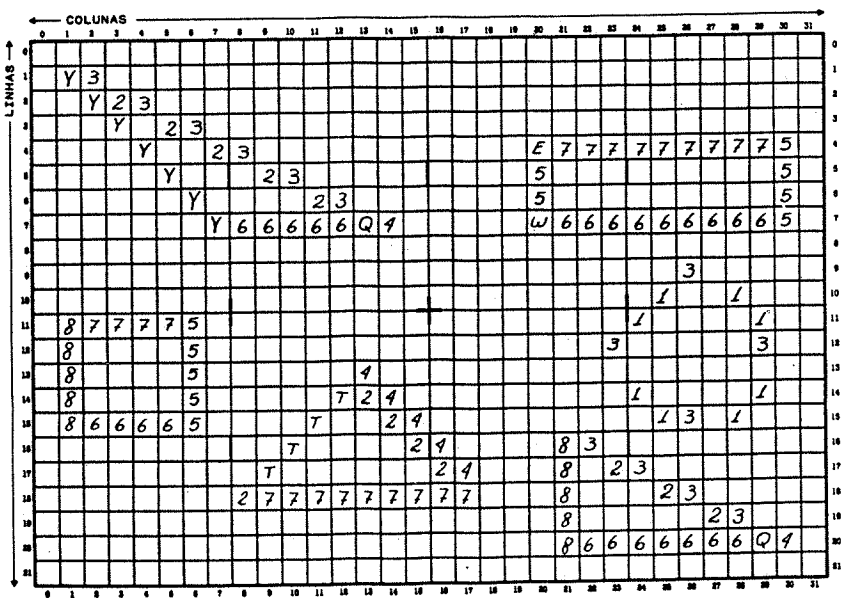

COLUNAS
LINHAS

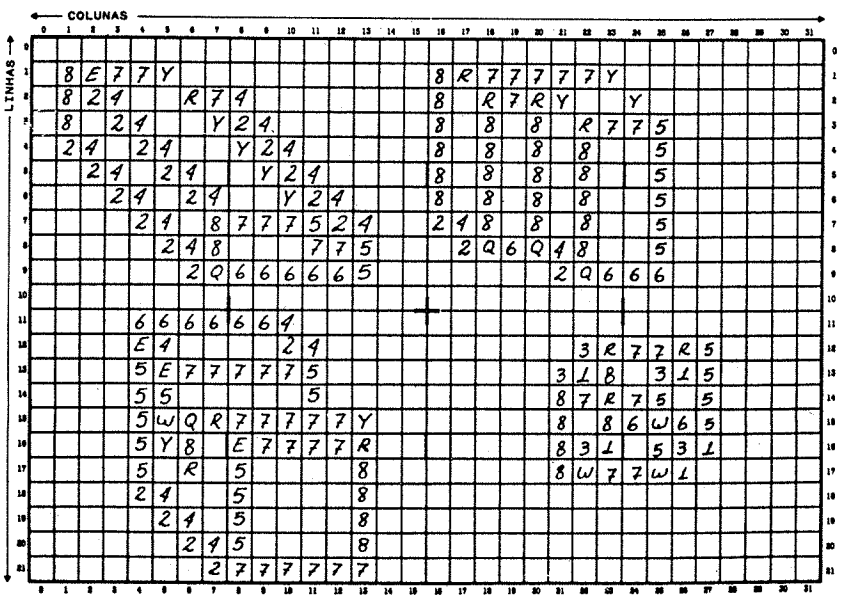

COLUNAS
LINHAS

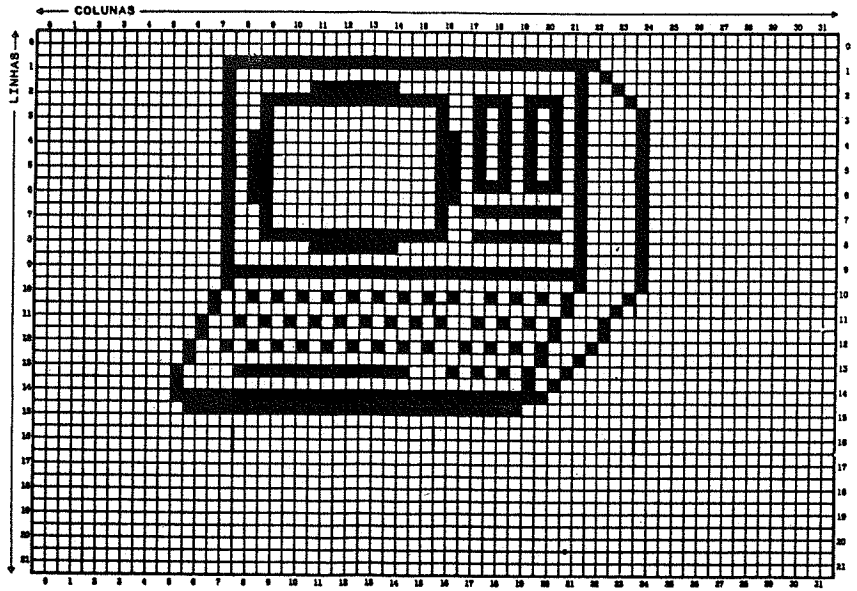

COLUNAS
LINHAS

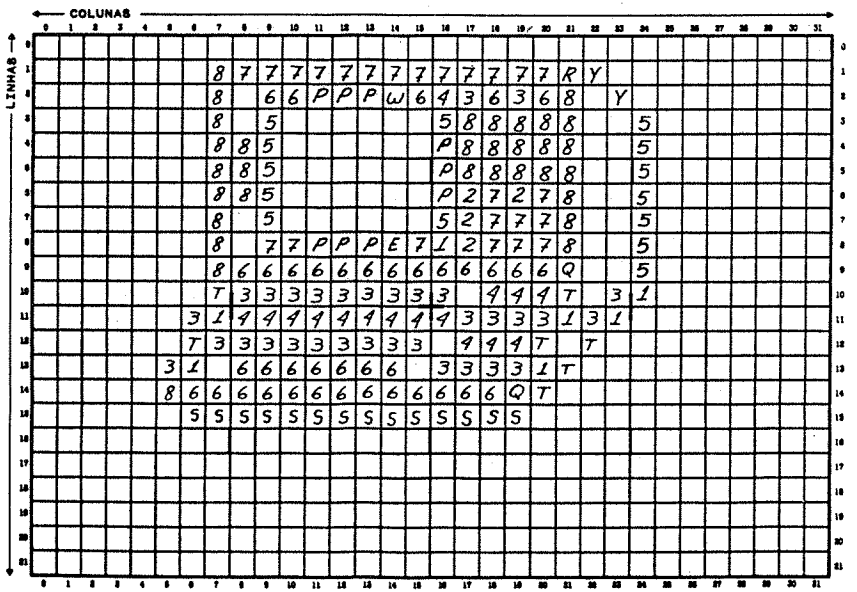

COLUNAS
LINHAS


COLUNAS
LINHAS


COLUNAS
LINHAS


COLUNAS
LINHAS





À

MICRON ELETRONICA COM IND LTDA

CAIXA POSTAL 100

12200 SAO JOSE DOS CAMPOS - SP

PEDIDO DE MATERIAL

Quant.	DISCRIMINAÇÃO	Unit.	TOTAL
		TOTAL + 10%	

Incluso cheque nº _____ c/banco _____

Remetente _____

End. _____

CEP _____ Cidade _____ Est. _____

SOFTWARE PARA MICROCOMPUTADORES SINCLAIR

MICRON SOFTWARE

O PROGRAMA ENTRE GRAPHICS EM CÓDIGO, USADO NESTE VOLUME, COM AS ROTINAS ADICIONAIS DA LISTAGEM 4 EM UM ÚNICO CASSETTE POR APENAS 1 ORTN.

ENTER GRAPHICS

Programa em código com facilidades de edição para imagens com caracteres gráficos. Armazena até 12 figuras de 22 linhas. Possibilita alterações futuras, apagar, listar, gravação e carga das imagens, independentemente do programa e ainda simula movimento, copiando instantâneamente as doze imagens para o vídeo.

A LISTAGEM 4

Inclusa no ENTER GRAPHICS, compõe-se de rotinas extremamente úteis para elaboração de jogos. Simulam movimento, rodam o vído para a esquerda ou direita, scrool de 1 coluna para cima (tiro) e inversor de vídeo.

BLOCO PARA DESENHO

Bloco para Mapeamento do Vídeo Sinclair.
50 folhas tamanho ofício com duas telas de 22 linhas por folha, por apenas 0,3 ORTN.

Apresenta, por folha, uma tela com 22x32 posições e outra de 44x64 posições. Contém ainda uma listagem ampliada dos símbolos gráficos com as teclas de localização.
As páginas da parte IV deste volume são deste bloco com redução de 45%.

SOFTWARE PARA MICROCOMPUTADORES SINCLAIR

ZX SOFTWARE

OS QUATRO MELHORES PROGRAMAS EM CÓDIGO, JAMAIS PRODUZIDOS PARA UM SINCLAIR EM UM ÚNICO CASSETE. POR APENAS 1,5 ORTN. TODOS PARA 16K DE RAM, COM SLOW

ASSEMBLER

Escreva os seus programas em código, usando diretamente os mnemônicos do Z80 com este Editor e Monitor de Assembly. Interpreta todas as instruções do Z80. Oferece excelentes facilidades de edição, manipulação do cursor, códigos de erro, entrada de texto, números, labels, repetição de tecla, etc...

DISASSEMBLER

Lê códigos no Assembly do Z80. Fornece os endereços em decimal, os códigos em hexadecimal seguidos dos mnemônicos completos. Interpreta todas as instruções do Z80.

COMPILER

Transforme instantaneamente os seus programas em Basic em programas em código, usando o Compiler. Aceita quase todos os comandos do Basic Sinclair e passa a rodar os programas até 50 vezes mais rápido.

MONITOR

Programa para estudo de programas em código. Permite listar a Memória, Registers e Flags. Pode-se ainda introduzir Breakpoints, converter Hex para Decimal, decimal para Hex etc...

Distribuído por:

MICRON ELETRÔNICA

PUBLICAÇÕES

MICRON ELETRÔNICA

PARA MICROCOMPUTADORES

sinclair®

SINCLAIR É MARCA REGISTRADA DE SINCLAIR RESEARCH LTD.

- **200 DESENHOS PARA TK E CP200**
  Mais de 200 desenhos para jogos e ilustrações, codificados.
  Programas para: edição dos desenhos codificados; cópia instantânea do Display, simulando movimento; transposição de vídeo entre programas, etc...
  Ensina as técnicas de vídeo, mapeamento, movimento, etc...

- **40 ROTINAS EM LINGUAGEM DE MÁQUINA**
  Rotinas para aumentar o desempenho de seus programas em Basic ou em Linguagem de Máquina. Soma, Subtração, Multiplicação e Divisão de strings e números binários. Scrool e rodar o vídeo em todas as direções, etc...

- **HARDWARE SINCLAIR**
  Led Monitor, Reset, Fonte, Superaquecimento, Monitor de Vídeo, Interfacing a PIO, Expansão interna de RAM de 2, 4, 8 e 16K, Portas in/out, Teclado, Lay-out dos micros Sinclair e dos circuitos impressos, incluindo esquema dos micros Sinclair ZX.

● 30 JOGOS PARA TK82 E CP200
Damas — Labirinto — Enterprise — Golfe — Velha — Visita ao Castelo — Cassino — Roleta Russa — Corrida de Cavalos — Vinte e Um — Cubo Mágico — Senha — Banco Imobiliário — Forca — Dados — Invasores — etc.

PROGRAMAS NO CÓDIGO DA MÁQUINA
Inversão de Vídeo — Som por Software — Labirinto — Destrava Soft.

● CÓDIGO DE MÁQUINA PARA TK E CP200
Números Binários e Hexadecimais — Arquitetura do Z80 — Editando em Código — Programa para Edição — As Instruções do Z80 em Exemplos — Sub-rotinas da ROM — A ROM de 8K — Dicionário das Instruções — HEX X Mnemônicos — Hex X Decimal — Incluindo os Programas: Scroll — Save Display no Ram Top — Contadores de Tempo ou Ponto — DataFile — Renumber — Labirinto — Som por Software — Micron Pao — Bombardeio, etc...

● DICIONÁRIO DO BASIC SINCLAIR
Para os micros ZX81, NEZ8000, CP200, TK82C e TK83, TK85, RINGO, AS1000, etc... Reune todas as Instruções — Funções — Operadores e Caracteres descritos e exemplificados em ordem alfabética. Inclui Divisão do Basic Sinclair — Convertendo outros Basics — Contando os Bytes — Economizando Memória.

● APLICAÇÕES SÉRIAS PARA TK85 E CP200
Quem é Sinclair — Convertendo outros Basics — Contando os Bytes — Economizando Memória — Fluxogramas — Top Down — Erros da ROM — Conhecendo a Impressora — Chaining Programas — Sub-Rotinas em Cassete — Folha de Pagamento — Balancete — Correção Monetária do Imobilizado — Das Contribuições do IAPAS — Contas a Receber — Cadastro de Clientes — Conta Bancária — Correção de Provas — Processador de Textos — Estatística — Custo — Orçamento Doméstico — Ramtoper em Código — etc...

● 45 PROGRAMAS PRONTOS PARA RODAR EM TK82C E NE Z8000
Arquivos — Estoque — Plano Contábil — Folha de Pagamento — Agenda Telefônica — Caça ao Pato — Trilha — Jogo da Velha — Forca — Dado Tabelas — Tabuadas — Conversão de Coordenadas — Média — Progressão — Tabela Price — Fibonacci — Depreciação — Renumerador de Linhas em Código — Etc...